Localized Crime Prediction Methods

A Dissertation

Presented to

the Faculty of the School of Engineering and Applied Science

University of Virginia

In Partial Fulfillment

of the requirements for the Degree

Doctor of Philosophy (Systems and Information Engineering)

by

Mohammad Al Boni

August 2017

Approval Sheet

This Dissertation is submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy (Systems and Information Engineering)

Mohammad Al Boni

This Dissertation has been read and approved by the Examining Committee:

Matthew S. Gerber

Donald E. Brown

Robert D. Mcconnell

Quanquan Gu

Hongning Wang

Accepted for the School of Engineering and Applied Science:

Craig H. Benson, Dean, School of Engineering and Applied Science
August 2017

Abstract

The convergence of public data and statistical modeling has created opportunities for public safety officials to prioritize the deployment of scarce resources on the basis of predicted crime patterns. Current crime prediction methods are trained using observed crime and information describing various criminogenic factors. Researchers have favored global models (e.g., of entire cities) due to a lack of observations at finer resolutions (e.g., ZIP codes). These global models and their assumptions are at odds with evidence that the relationship between crime and criminogenic factors is not homogeneous across space. In response to this gap, this dissertation presents a framework for building localized crime prediction models. The proposed models achieve localization using three approaches: 1) quantifying micro-level daily routine features from social media; 2) building area-specific models at finer resolutions (e.g, neighborhood), and 3) proposing a new way for estimating historical crime density at the local level. Experimental results on real crime data from Chicago, Illinois indicate predictive advantages over multiple state-of-the-art global models. Furthermore, this dissertation includes a comprehensive performance analysis of existing evaluation metrics and a human factor study on the effectiveness of visualization techniques for making decisions that are informed by forecasts. The dissertation follows a holistic approach for crime prediction modeling such that it is not only important to build high quality models but also properly evaluate them and effectively visualize their outcomes. This research helps security agents to better allocate their resources and people to better manage crime risks, which ultimately improve public safety.

To the innocent victims of the Syrian revolution, women and men, young and old.

Acknowledgments

I would like to thank my advisor, Professor Matthew S. Gerber, who has created a great environment for me to learn and grow. I am fortunate to have had professor Gerber as my advisor. He has guided, inspired and motivated me throughout my time under his supervision. He has taught me how to become an independent researcher while always offering to help me overcome any challenges facing me with research. To my parents, Mouwafak and Ibtesam, your endless love and support has been the driving motive for me to work hard and succeed. Nothing would make me happier than making you feel proud. To my wife, Dana: thank you for everything you did and said for the last sixteen months. Every day since I have known you, I have become a better and a happier person. To my siblings, colleagues, friends, and everyone else: thank you for believing in me. **Thank you all**.

Contents

List of Tables

List of Figures

Introduction

Crime has ever been a major problem as it has a big social and economic impact on people's daily life. Starting early 90s, crime rate peaked at 5,865 crimes per 100,000 people in the United States, and since then it has been declining [1]. In 2016, the overall crime rate in the United States was about half of its value in 1991 reaching 2,857 crimes per 100,000 people [1]. The decline in crime rates has been widely debated. Scholars has attributed the decline to various factors such as economic growth, increased incarceration, etc. [2–4]. In part, the decline was caused by advancements in proactive and prevention policing strategies. Considering the limited resources that law enforcement agencies have, reliable crime predictive modeling has become an essential part of effective proactive policing. Such predictive models can help the police to make decisions on where and when to allocate resources.

Crime prediction refers to the process of identifying areas with highest chance of future offenses, known as hotspots, and by allocating resources to these areas, the police hope to deter the potential offenses. Using retroactive analysis, practitioners can evaluate forecasting models by counting actual incidents occurred in hotspots. Assuming that if the police were to patrol those areas, they would have a chance in preventing those crimes.

Overtime, predictive modeling of crime has been developed and many advancements in terms of the theory and the algorithms have been introduced. Major milestones in the development of crime prediction models include:

- **Hotspot models:** the intuition and main premise of hotspot models is that areas with previous crime records will continue to observe future crimes. Therefore, the more frequent the incidents are, the higher the future threat become.

- **Temporal and time-series models:** taking into consideration not only the location of historical incidents but also their occurrence time allowed for better capturing crimes

with time-related patterns (e.g., day vs night, weekday vs weekend, winter vs summer, etc.)

- **Spatio-temporal and risk terrain models:** crime has remained a complex phenomenon, and historical offenses are not always sufficient for forecasting future crimes. This challenge motivated researchers to include various variables and factors characterizing the social and physical environment (e.g., demographics, risk factors, locations of police stations, etc.)

- **Social media:** the emerge of social media as source of live and big data about an environment inspired researchers to build prediction models that include variables characterizing the dynamics of that environment.

A great progress has been made, yet most existing models fall short in addressing an important aspect of crime: **heterogeneity**. By building a single model to forecast future incidents in an entire area of interest, researchers have ignored the diverse nature of crime and how its patterns vary from one location to another. I address this gap by proposing a framework for building localized crime prediction models. In this framework, localization can be achieved by three approaches: 1) designing variables that measure the micro-level differences between areas, 2) building area-specific models, and 3) introducing learning algorithms that are informed by local and neighboring information. Experimental results showed how localized crime prediction models can attain substantial gains over traditional global models. Furthermore, I follow a holistic approach for crime prediction modeling and I study two additional and complementary concepts: 1) evaluation metrics and 2) visualization. It is important not only to build high quality predictive models, but also to evaluate them properly and visualize them in a way that allow practitioners to make the most effective use of their outcomes.

This dissertation is organized as follows. In Chapter 2, I review the literature on crime prediction modeling and I give examples of models that cover the milestones mentioned above.

In the same chapter, I present the formulation of basic spatio-temporal global crime prediction models. Chapter 3 begins by introducing the theoretical framework for build localized crime prediction models. The chapter then provides a detailed description and evaluation of four models that can are realization of different angles of the localization framework. Chapter 4 begins with a comprehensive performance analysis on existing evaluation metrics. As part of this analysis, I examine the effect of changing the problem settings on models' performance. The chapter then presents extensions of the evaluation metrics and provide recommendations on how to design a proper and fair evaluation of forecasting models. Chapter 5 presents a human factors study on map visualization methods. Chapter 7 describes a case study of using localized crime prediction models in a real world problem: the NIJ real-time crime forecasting challenge. I conclude, in Chapter 7, with a summary of contributions and future work.

Global Crime Prediction Models

2.1 Related work

Researchers have developed several statistical methods for crime analysis and prediction [5, 6]. Kernel density estimation (KDE) is a common technique used to compute a retrospective visualization of crime concentration (often called a hot spot map) [5]. KDEs are non-parametric models that, when applied to crime, estimate higher risk in areas of higher historical density. KDEs have been widely adopted by police analysts for hotspot mapping. It is also included in many commercial and non-commercial spatial analysis packages such as ArcGIS, MapInfo, R, and CrimeStat III. As a predictive method, KDE assumes that areas with historically high crime rates are more likely to be victimized in the future than areas with historically low crime rates. The advantages of KDE are that it is simple to implement and applicable in any area with a recorded history of crime locations. However, KDEs do not benefit from the abundance of information (e.g., geospatial and social media databases) that might explain and predict the occurrence of crime. RTM attempts to address this limitation of KDEs by explaining the occurrence of crime in terms of various physical and demographic risk factors. Each factor is measured by a risk score, and the RTMs for different crime types emphasize different risk factors. For example, when analyzing sexual assaults, RTMs consider risk factors such as proximity to bars, clubs, and schools, and the distribution of age, gender and wealth [6].Furthermore, the RTM framework can perform a hot-spot analysis by adding a layer for the historical crime records. After measuring these risk factors, the RTM combines all layers and generates a risk score indicating the relative likelihood of future crime across the study region. Other crime prediction models by Brown and his colleagues have investigated the temporal aspect of crime and incorporated ideas from random-utility theory to model the preferences of criminal offenders [7, 8]. Liu and Brown developed an event prediction framework using point process models [7]. In this framework, the authors included three

types of data: location and times of previous criminal incidents, and a set of features that represent preferences of criminal offenders with respect to alternative locations. Smith and Brown developed a multinomial choice model using the Type-1 extreme value distribution [8]. The authors did not restrict their model to a specific type of feature, and they tested their approach using crime data from Richmond, Virginia, USA using distance measurements and demographic features. Although hot-spot, RTM and other crime prediction models have effectively modeled crime likelihood in different areas considering their surrounding social and physical environment, these models did not make use of the social media textual content that reflect the dynamics in the surrounding environment. To account for the rich and the rapidly growing social media sources, researchers have proposed various approaches for crime prediction based on topic modeling [9, 10]. In [9], the authors extracted different hidden topics using a latent Dirichlet allocation (LDA) algorithm. Then these topics were included as predictors in a logistic regression model. The authors have showed that this model improved the hit-and-run prediction in Charlottesville, Virginia, USA. This model was extended in [10] to include the density of crimes along with the latent topics. Using this model, the authors were able to improve the prediction for 19 out of 25 crime types that occurred in Chicago, Illinois, USA.

2.2 Mathematical approach

I treat crime prediction as a classification problem, where the units of classification are spatial points p and the response is binary, indicating the odds of observing a crime at point p. In another words, the global model estimates the relative risk of crime type T at point p using a set of predictor features.

To build my global crime prediction models, I first discretize the geo-spatial surface of an area of interest. I create a grid of points with a fixed cell size. Each of these points is labeled *NONE* (for the non-occurrence of crime). Then, I create points from the locations of all known crimes of type T, and I combine these points with the *NONE* points. In cases where a *NONE* and T point coincide, I remove the former. Next, I use all points (*NONE*

and T points) to train a binary classifier with the following form:

$$Pr\left(Label_p = T | f_1(\theta_p), ..., f_n(\theta_p)\right) = F(f_1(\theta_p), ..., f_n(\theta_p)) \tag{2.1}$$

where $f_1(\theta_p), ..., f_n(\theta_p)$ are features describing point p with parameters θ, and F is a link function relating the predictors to the response. In this work, I used the logistic function:

$$F(f_1(\theta_p), ..., f_n(\theta_p)) = \frac{1}{1 + e^{-\left(\beta_0 + \sum_{i=1}^{n} \beta_i * f_i(\theta_p)\right)}} \tag{2.2}$$

In words, the global model contains coefficients representing the relationship between (1) crime occurrence and non-occurrence at point p and (2) various features of p. These coefficients apply uniformly to the entire city. This is a standard model configuration, and I use it as a baseline. Also, this formulation allows for building a wide range of models by quantifying appropriate features. Next, I present three type of features that represent historical crime records and spatio-temporal patterns. These features were included in a previously developed crime prediction system [11], and used in previous works such as [12] and [10].

2.2.1 Kernel density estimation

Hotspot mapping is tool used by police departments to analyze historical crime records and identify future areas of high risk. According to a report by the Bureau of Justice Statistics in 2007, 92% of police departments serving more than 1 million citizen in the United States use hotspot mapping [13]. Hotspot mapping assumes that criminal activities are spatially stable over time. To the extent that this is true, hotspot methods can use historical crime incidents to generate predictions for future areas of high risk. There are many ways to generate hotspot maps [5, 14–17]. A recent study has shown that kernel density estimation (KDE) outperforms other hotspot methods in terms of prediction accuracy [18].

KDEs are models that, when applied to crime, estimate higher risk in areas of higher historical density. The earliest KDE formulation was put forth by Rosenblatt in 1956 [19]. It

was later extended by Parzen in 1962 [20]. After building the spatial grid, I calculate density estimates for the grid points such that points with more surrounding incidents have higher density estimates and therefore higher predicted risk. The distances between incident points and the grid points are scaled using a bandwidth parameter, and the results are fed to an interpolation function. The KDE is formally defined as

$$f(\theta_p = \{p\}) = k(p, h) = \frac{1}{Ph} \sum_{j=1}^{P} K\left(\frac{||p - p_j||}{h}\right), \tag{2.3}$$

where P is the total number of crime incidents, h is a smoothing parameter (bandwidth), p is the point at which a density estimate is calculated, $||\cdot||$ is the L-2 norm, and K is an interpolation (kernel) function. Also, Equation 2.3 can be modified to allow for estimating a temporal density estimates by using $\theta_p = \{p, t_1, t_2\}$ such that P is the total number of crimes of type T that occurred between times t_1 and t_2.

KDE requires the analyst to define a number of parameters: the bandwidth, the kernel function, and the grid cell size. Researchers have suggested different values for the bandwidth. Bailey and Gatrell claimed that the bandwidth should be set to 0.68 times the total number of incidents raised to the power of -0.2 [21]. Felson argues that the bandwidth should be set to cover a few blocks surrounding each point [22]. Most statistical packages offer default values for the bandwidth based on heuristics. For example, ArcGIS provides five different algorithms for setting the bandwidth.[1] One approach is to set the bandwidth to the average distance between the center points and all incident points. R software, on the other hand, provides a function[2] to estimate the bandwidth using a multivariate selection algorithm [23]. As for the interpolation method, there are a number of functions that can be used such as normal, triangular, quadratic, uniform, and bi-weight density functions. However, there has been no study of which function should be used in different situations, and often this is

[1] Default search radius (bandwidth) algorithm. Retrieved July 18, 2016, from http://pro.arcgis.com/en/pro-app/tool-reference/spatial-analyst/how-kernel-density-works.htm.

[2] Hpi function, ks package, p10. Retrieved July 18, 2016, from https://cran.r-project.org/web/packages/ks/ks.pdf.

determined empirically. Another parameter, which is not formally part of the KDE formula, is the grid cell size for the hotspot map. Chainey and Ratcliffe have suggested that the grid cell size should be set to the shorter of the width and height of the region's bounding box divided by 150 [15]. Caplan and Kennedy suggest that the optimal grid cell should be within 1/2 and 1/3 the mean blockface length [24].

Another important factor is the computational complexity of KDEs. The summation in Equation 2.3 is over all P incident points, and this summation is calculated for all G grid points. Thus, the L-2 norm, the smoothing calculation, and the kernel function will be evaluated $O(G * P)$ times to produce a hotspot map. In a city like Chicago, Illinois, it is not uncommon for G to be on the order of 10^4 and P to be on the order of 10^3 for just one month of crime within the city boundary. The resulting computational burden can be high, with $O(10^7)$ evaluations of the kernel function. For example, in the experiments described later in this chapter, I build a grid of 15,574 cell points, and for one month of training theft incidents, I would have around 4,500 incident points. Therefore, assuming that the bandwidth is given, a KDE operation would require around 140 million non-basic operations (i.e., L-2 norm and kernel function). In practice, the incident data are downsampled to reduce P and produce a feasible execution time at the expense of losing data.

2.2.2 Spatial features

Spatial features allow analysts to incorporate the characteristics of the physical and social environment surrounding grid points. In this section, I consider three types of spatial features: distance, density, and attribute features. Distance features estimate the linear distance (i.e., euclidean distance) from p to the entity (e.g., linear distance to a major street). Density features estimate the spatial density of the physical entity at p, as measured by yet another KDE (e.g., the value at p of the KDE built from police station locations). Finally, attribute features are used to quantify spatial and/or social categorical or numerical values at p (e.g., the ZIP code that covers p or the percentage of females, as reported in the census, at p). Spatial features tend to be time invariant, and are parameterized by $\theta_p = \{p\}$. However, in

some cases, and when data is available, spatial features can change over time. Similar to temporal KDEs, spatio-temporal features take $\theta_p = \{p, t_1, t_2\}$ as parameters such that the feature values are quantified between times t_1 and t_2 (e.g., the percentage of people at home at point p between times t_1 and t_2).

2.2.3 Temporal features

The temporal component of the global prediction model is captured by two features, sine and cosine, which quantify the temporal position of the point p within a 24-hour cycle. I divide each day into four six-hour slices: 12:00am-5:59am, 6:00am-11:59am, 12:00pm-5:59pm, and 6:00pm-11:59pm. Figure 2.1 illustrates my time-slice approach. Both features take only one parameter $\theta_p = \{t_c\}$, where t_c is time within the current slice. Formally, temporal features are $\sin(\zeta(\theta_p = \{t_c\}))$ and $\cos(\zeta(\theta_p = \{t_c\}))$ such that

$$\zeta(t_c) = \frac{2 * \Pi * t_c}{t_\zeta}, \tag{2.4}$$

where t_ζ is the number of hours within the current period (24 in my models). Thus, $\zeta(t_c)$

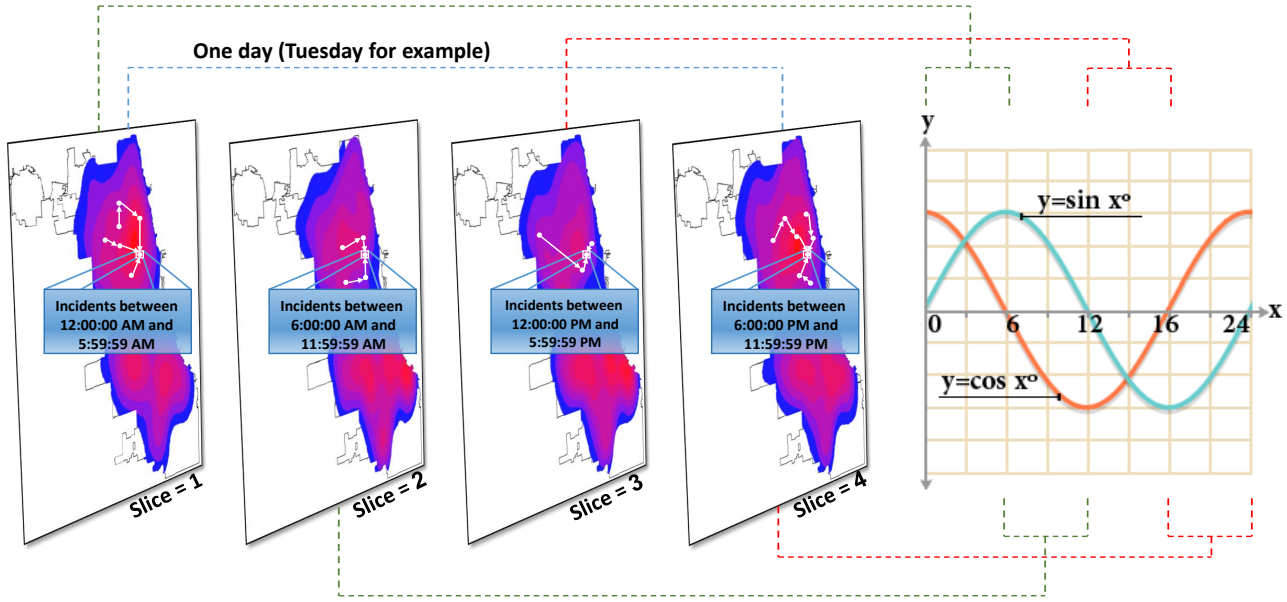


Figure 2.1: An illustration of my approach for dividing each day in the training and prediction into four six-hour slices within which I extract different crime and routine features. The graph on the right represents the sine and cosine features associated with each slice.

is a function that measures temporal position of point p within a 24-hour period, with measurements at the start and end of the period having the same values. For example, Slice 1 in Figure 2.1 occupies the $0\% - 25\%$ interval of the day. A similar numerical representation of temporal position would indicate that Slice 4 occupies the $75\% - 100\%$ interval. However, these numerical representations do not match the intuition that the end of Slice 4 (100%) is identical to the start of Slice 1 (0%). The sine and cosine features transform the slice positions to better account for this intuition. The sine feature reflects the difference in crime distribution between morning and afternoon (see the sine curve in the right of Figure 2.1) while the cosine feature captures differences between day and night (see the cosine curve in the right of Figure 2.1).

2.3 Experimental setup

I compute the feature values for all points mentioned in section 2.2 ($NONE$ and T). Then, I use a machine learning package, such as Weka [25], LibLinear [26], etc., to train the binary classifier in Equation 2.1 on all points in the training window. To make a prediction for a future time window, I replicate the coordinates of the $NONE$ and T points (not the labels) within each future time slice, extract feature values for these points, and apply the fitted model to these feature vectors to obtain predictions for each future testing window. Since I do not know what the future value will be for some features such as the KDEs, I lag all features by a certain window, 7 days for example, so that all predictions are derived from data that would be available in a practical application of my method. The generated predictions, also called threat surface and an example of which is shown in Figure 2.2, can be later used to assist the police in making resource allocation decisions such that more resources would be allocated to high risk areas, i.e., areas with a high likelihood of crime occurrence. Furthermore, I repeatedly train and predict, moving the model ahead by a fixed number of days each iteration. This setup emulates the use of my model in a practical situation where we have historical crime records and we want to predict crimes for the following days. This setup will be used throughout this work. However, in different chapters, I decided to vary

the size of training and testing windows. I also use different variations of the feature types presented earlier. The main motivation behind doing so is that in real world scenario, we might have limited access to spatial and/or temporal data. Therefore, I wanted to show the performance of my models in a more realistic setup.

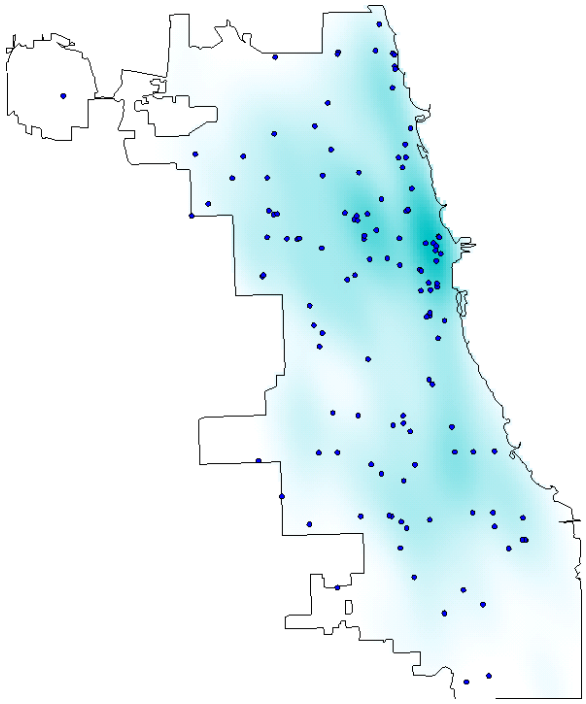


Figure 2.2: An illustration of threat surface generated by a crime prediction model. In this illustration, I show the map of Chicago, Illinois with assault incidents shown as blue points. The heat map represents the likelihood of assault occurrence such that darker color means higher likelihood. Note that zero-padding is required for grid cells on the boundary.

2.4 Evaluation

There are a number of metrics for evaluating crime prediction models (as I will discuss later in Section 4). For this work, I use surveillance plots to evaluate different prediction models, an example of which is shown in Figure 2.3. A surveillance plot shows the proportion of true predicted burglaries (y-axis) that occur within the most threatened area predicted by the model (x-axis). Figure 2.3 shows that around 60% of future crime incidents (y-axis) can be observed by surveilling the top 35% most threatened area (x-axis), as predicted by the model. The area under the curve (AUC) provides a scalar summary for the surveillance curve.

A detailed comparison between surveillance plots and other metrics will be later discussed in Chapter 4.

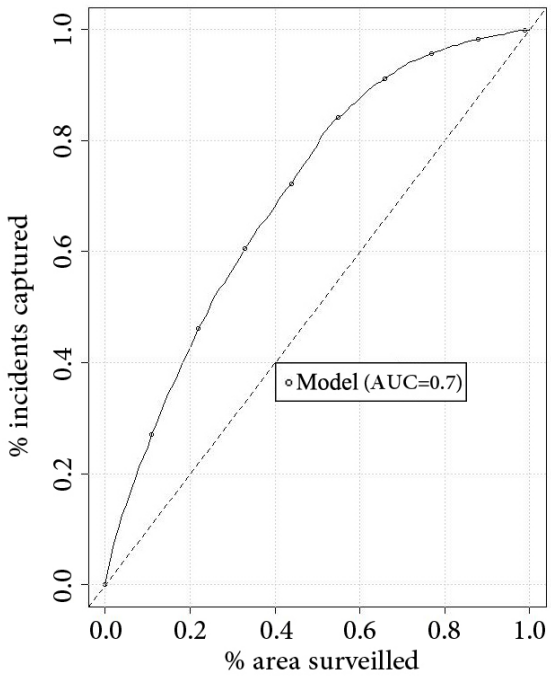


Figure 2.3: A surveillance plot, which shows the proportion of true future incidents (y-axis) that occur within the most threatened area predicted by the model (x-axis). This surveillance plot is summarized by its area under the curve (AUC) value.

Description	Type
Major Streets	Distance
Police Stations	Distance
Police Stations	Density
Hospitals	Distance
Hospitals	Density
Bicycle Racks	Distance
Bus stops	Distance

Table 2.1: Spatial features included in sample theft prediction models. Distance features indicate linear distance from an analysis point to the spatial entities, whereas density features quantify the spatial density of those entities at that point.

To illustrate the use of global models with different features, I setup a sample prediction problem of real theft incidents from Chicago, Illinois, United States. I divided the city of Chicago into cells of size 500 meters. I set the size of the training window to 7 days

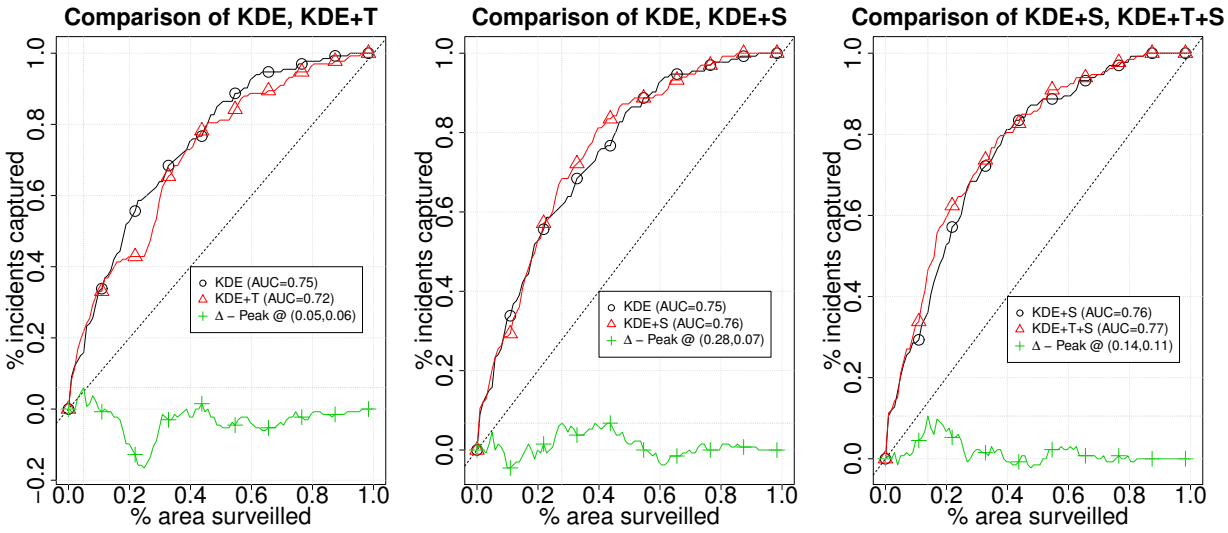


Figure 2.4: Theft prediction performance using global models. Surveillance plots using five series: Density (KDE), Density+Temporal (KDE+T), Density+Spatial (KDE+S), Density+Temporal+Spatial (KDE+T+S), and the gains from adding additional features (the location of peak gain is indicated with Δ-Peak).

(1/1/2014-1/7/2014) and the prediction window to 1 day (1/8/2014). During January 8^{th} 2014, there were 133 thefts constituting 22.35% of all crime. I compared the performance of four prediction models: one that uses the crime density feature, a second that adds the temporal sine and cosine features to the first model, a third that adds seven spatial features, shown in Table 2.1, to the first model, and a fourth that contains crime density, temporal and spatial features. Figure 2.4 shows the surveillance plots for the sample theft prediction experiment. The plots contain five series: "KDE", "KDE+T", "KDE+S", "KDE+T+S", and "Δ-Peak @(x,y)". The first four series correspond to the four prediction models respectively. The Δ series shows the gains from adding additional features to the model. The peak difference is indicated with "Peak @(x,y)". Figure 2.4 shows more gain from adding spatial, rather than temporal, to the KDE. Note, in Figure 2.4, not all of the surveillance area (the x-axis) is equally important. Since the police have limited resources, they cannot surveil all threatened areas. Therefore, the closer the peak gains are to $x = 0$, the more important they become to the police. For example, comparing the area under the curve (AUC) for KDE and KDE+T would suggest that temporal features harm the prediction performance

(0.75 using KDE vs 0.72 using KDE+T). However, the police would still prefer the latter model since within a practical surveillance area (5% or about 11.7 square miles), adding temporal features would improve the accuracy by 6%. In words, if the police would surveil the most threatened 11.7 square miles as forecasted by a model with crime density and temporal features, they would have a chance in preventing 6% more thefts than they would if they surveil the most threatened 11.7 square miles as forecasted by a model with only crime density feature. Overall, a model with crime density, temporal and spatial features provides the best performance. Nonetheless, prediction performance varies from one area to another, from one time to another, and from one crime type to another. Therefore, it is important to explore various types of features when modeling for different situations.

2.5 Conclusion

In this chapter, I presented the mathematical framework of a standard global crime prediction model. With available data, these models enable crime analysts to capture the spatial and temporal characteristics of an area of interest, and generate threat surfaces. These surfaces can be used by practitioners to optimize resource allocation, act proactively to prevent crime, and improve public safety. However, these global models are estimated using information aggregated from many smaller areas. The models assume that the smaller areas are homogeneous with respect to crime occurrence and the various criminogenic factors studied. These models and their assumptions are at odds with evidence that crime is not homogeneous across space [27]. This gap raises the following research questions: (1) Is it possible to design and estimate features that would capture micro-level differences between areas? (2) Is it possible to build area-specific predictive models using sparse data? (3) Is it possible to change the learning approach to take into consideration the heterogeneity of crime patterns across different areas, and (4) Will these localized approaches introduce a gain in performance over the global models, supporting hypotheses about non-homogeneity of crime? I address these questions in the remainder of the work. My experiments provide an affirmative answer to each question.

Localized Crime Prediction Methods

3.1 Theoretical framework

It has been a constant driving motive for researchers to build more accurate, reliable and robust predictive models. Especially when predicting human behavior, models have failed to perform perfectly on all types of individuals and/or behaviors. The challenge remained in capturing and learning distinct, yet not random, behavioral patterns. To deal with this challenge, researchers have developed a wide range of approaches to improve models' performance on individuals or group of individuals. These approaches have been used in many applications such as web search and web mining [28–30], opinion and sentiment analysis [31, 32], marking and recommendation [33, 34], education [35], or even health [36, 37]. Depending on the problem domain, these methods can be called personalization, customization, adaptation, and in the domain of spatio-temporal modeling, localization. These methods can be defined as the process of adapting or tailoring a model or a product to be a better fit for an entity (e.g., a person), or a group of entities.

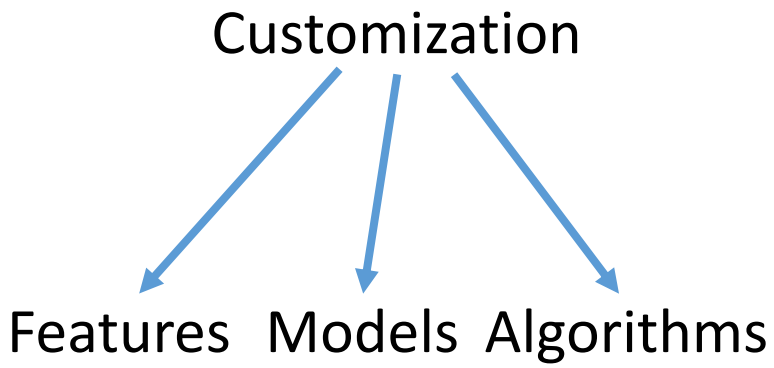


Figure 3.1: Theoretical framework for building customized models through designing feature values, machine learning models, and/or training algorithms.

In this section, I outline a theoretical framework, shown in Figure 3.1, under which customization approaches, the ones mentioned above and others, fit. Customization can be

achieved through three different means: 1) customizing feature values to entities, 2) building separate models for different entities, and 3) adjusting learning algorithms to take into consideration entity-based information. The first and widely used approach is to design and quantify features that characterize entities. This allows the training of models to be informed by the unique differences between those entities. For example, including an age feature in a linear regression model predicting employees' productivity will allow the model to distinguish between various age groups. Assuming that the age feature has a non-zero coefficient in the model, its value with contribute (positively or negatively) to the final productivity estimate. The limitation of this approach is that the relationship between entities and features are determined by a single model, and therefore, all entities will have the same relationship (e.g., model coefficient) regardless of their inter differences. In this second approach, separate and unique models are fitted for various entities. While this approach guarantees a unique relationship between models and entities, it requires an amount of observations sufficient to achieve a good fit. Researchers have addressed this challenge using various methods such as transfer learning [38], model adaptation [31, 39] collaborative learning [40, 41] etc. Finally, in the last approach, a single model is trained for all entities, yet the training algorithm will not only aim to minimize the error but also preserve entities distinctive characteristics during the training.

In the context of crime prediction, an entity can be a street, a neighborhood, a zip-code, etc. and the task is to build localized models for those entities. Crime varies from neighborhood to neighborhood [27, 42]; however, current crime prediction models have ignored this variation, favoring global models (e.g., of entire cities) due to a lack of observations at finer resolutions (e.g., ZIP codes) [5–10, 43, 44]. For example, a single global model might estimate the correlation between spatial density of bus stops and the occurrence and non-occurrence of thefts. Public safety officials might use a positive correlation between these variables to justify the allocation of resources (e.g., police patrols) to areas with the highest density of bus stops. However, this global correlation might not hold within each sub-area. The true

correlations might be positive in one sub-area and negative in another. In this chapter, and in accordance with the localization theoretical framework, I address this challenge in three different ways. I 1) estimate micro-level daily routine features that quantify daily routine movements of individuals from social media; 2) build area-specific models that are trained at finer resolutions (e.g., neighborhood), and 3) propose a new way for estimating historical crime density at the local level. The first and third approaches allow for capturing the dynamics of the prediction problem at a local level even though the trained model is a single global model. As for the second approach, each local area will have its own separate model.

3.2 Micro-level routine activity features

3.2.1 Introduction

Human mobility studies originated within urban planning research as well as within sociology [45], networking [46, 47], mobile computing [48, 49] and epidemiology [50–52]. Research suggests that these movements are not random [53–56]; rather, people tend to move frequently between the same places, creating regular patterns similar to Lévy flights [53], an example of which is shown in Figure 3.2. In these patterns, people move between different clusters of places. The distance between points within a cluster tends to be small, and the distance between points in different clusters tends to be large. For example, an individual working at location X will probably have lunch at a nearby restaurant. This individual will also probably go to restaurants near his or her home. As a result, this individual will have two clusters of places: one for places around work and one for places around home. The fact that most people spend at least third of their time at home [57] makes the relationship between human mobility and home locations particularly interesting. According to the United States Department of Labor, Bureau of Labor Statistics, individuals aged 15 and up sleep an average of 8.3 hours per day [57]. However, not all individuals are sleeping and active at the same time. Most people work during the day and sleep at night. The movements of these people will peak during day hours (usually between 9 AM and 6 PM). On the other hand, when individuals work at night and sleep during the day, most of their movements will occur during the night (usually between 11 PM and 8 AM). This diversity in activity patterns creates challenges for automatically analyzing human mobility.

Cohen and Felson associated increasing crime rates during late 1960s with various changes in social structures and individuals' daily routines, which increased the number of suitable targets and decreased the presence of capable guardians [58]. Motivated by the regularity of individuals' daily routines [53, 55] and the hypothesized relationship between these routines and the occurrence of crime, I developed a method that quantifies daily routines using social

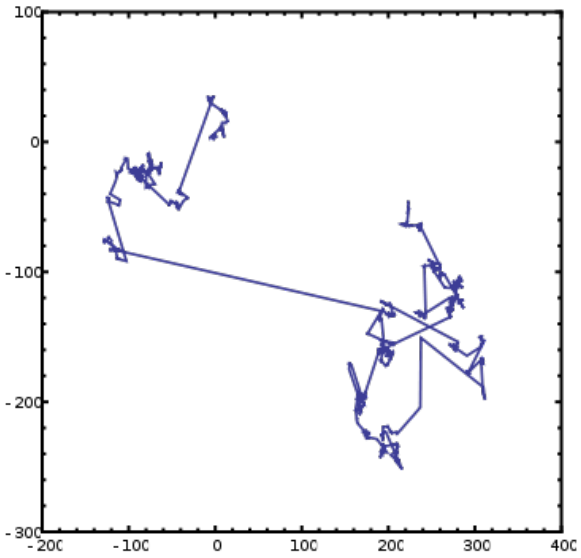


Figure 3.2: An example of the Lévy flight pattern that represents human movements in two dimensional space starting from $\{x, y\} = \{0, 0\}$. This example illustrates the fact that people do not randomly move between different places (e.g., home and work).

media and uses these quantifications as inputs to a statistical crime prediction model. As Felson and Cohen have observed, the spatial and temporal characteristics of daily routine activities (including legal ones) are key indicators of the type and intensity of crime in an area [59]. I hypothesize that social media will enhance the spatial and temporal analysis of past crime and produce improved predictive models for future crime.

According to routine activities theory, crime is driven by three factors: 1) motivated offenders, 2) the presence of a suitable target, and 3) the absence of capable guardians [58]. The theory predicts that crime is most likely to occur where and when these factors converge. Cohen and Felson developed this theory to explain the increase in crime rates during late 1960s. Although that decade marked an era of prosperity when wages were increasing, unemployment rates declined significantly, the number of people entering college increased, and the number of individuals living below poverty line decreased from 11.3 million to 8.3 million, crime rates were increasing [58]. Cohen and Felson focused only on crimes that involved direct contact between the offenders and the victimized targets (e.g., physical assault). They called these crimes direct-contact predatory violations [58]. For example, a

highly intoxicated person leaving a bar in the middle of the night and walking alone in a small, dark neighborhood, will create opportunities for criminals to commit offenses such as theft or assault.

3.2.2 Mathematical approach

In this section, I discuss my approach for correlating crime with micro-level daily movements, which can be divided into two parts. In the first part, I map individuals' tweets to the physical environment. Then, I reconstruct individuals' daily routines using a novel method for activity analysis. In the second part, I design a set of features for the daily activities using a bag-of-venues representation. Finally, I build a binary classifier for crime occurrence or non-occurrence that considers the historical crime density, various temporal components, and individuals' daily activity features.

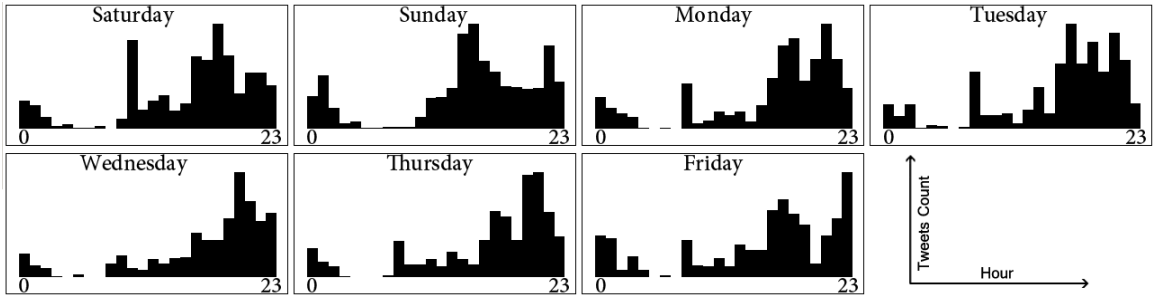


Figure 3.3: An illustration of day-by-day activity diagrams built for user 1007622991.

Reconstructing routine activities

The first step in reconstructing individuals' daily activities is to map individuals' posts to physical venues (e.g., restaurants and offices). I used Foursquare to map tweets to the physical environment. Although tweets are provided with latitude and longitude coordinates, my approach requires logical place names that abstract these coordinates. I used the Foursquare API[1] to collect 224,124 different logical places, called venues, for Chicago. These venues were listed under 579 categories such as restaurants, bars, and bus stops. The Foursquare API also provides a hierarchical list of categories with 10 at the top. Table 3.1 shows the venue counts for Chicago. Using the venues data set, I link each post to the nearest venue with a maximum

[1]Foursquare API. Retrieved from https://developer.foursquare.com/ .

distance of 5 meters. For example, if a user posted a tweet within 5 meters of a restaurant, I assume that the user was present at that restaurant at the given time. All posts that are not geo-located within 5-meters of a venue are excluded from the analysis. Next, to deal with the temporal heterogeneity of individuals' lifestyles, I developed a method of inferring when individuals will start their daily routines. The method is based on the observation that individuals become inactive during sleeping times. Motivated by this regularity, I developed an algorithm to identify sleeping times and reconstruct daily routines based on these sleeping times. In this approach, I aggregate the frequencies of tweets in each hour of each day and build seven different activity diagrams (one for each day of the week). These activity diagrams represent the activity of individuals on different days of the week. Figure 3.3 shows activity diagrams for user 1007622991. Next, I estimate the point with the least activity on each day and use this as a split point for the posts on that day. Once posts are split into subsequences, I represent them using the vector space model (VSM) [60]. In the VSM, each venue category defines one dimension. In this case, I used a 10-dimensional space to represent the 10 top-level venue types in the Foursquare venue ontology. Each route is a vector $r = \{w_1, w_2, ..., w_{10}\}$ where w_i is the importance of venue type i of the route r. Venue importance is calculated according to a Venue Frequency-Inverse Route Frequency (VF-IRF) weighting scheme (i.e., TF-IDF in the context of venues and routes instead of terms and documents [61]).

Example: Let u be a user that posted 8 tweets on a certain day. Using u's activity diagram, I split these posts into two routes, i.e., two subsequences before and after the user's least activity point: r_1 and r_2 with 3 and 5 tweets respectively. Next, let r_2 be: home→The Common Cup→CTA Bus Stop 1006→CTA Bus Stop 1120→Lincoln Building. I obtain the top-level venue category of each venue. For example, the "The Common Cup" is listed as a coffee shop venue which is itself listed as a food venue. The venue route for r_2 is: Residence→Food→ Travel & Transport → Travel & Transport → Professional & Other Places. I then calculate the importance vector for the route using the VF-IRF weighting scheme.

Venue category	Frequency(%)
Residence	74,310 (33.16%)
Professional & Other Places	39,258 (17.52%)
Shop & Service	33,983 (15.16%)
Travel & Transport	21,884 (9.76%)
Food	20,322 (9.07%)
Nightlife Spot	11,312 (5.05%)
Outdoors & Recreation	9931 (4.43%)
Arts & Entertainment	8310 (3.71%)
College & University	4459 (1.99%)
Event	355 (0.15%)
Total	224,124

Table 3.1: Frequency of venues in Chicago grouped into 10 different categories.

The VF values for the venues, shown in Table 3.1, are $1, 1, 0, 2, 1, 0, 0, 0, 0, 0$ respectively. Let the IRF values for the Residence, Food, Travel & Transport, and Professional & Other Places categories, which are computed across all routes (similar to a corpus in traditional IRF scoring), be 1.84, 1.64, 1.7, and 1.57 respectively. Then, the VSM representation for $r_2 = \{1.84, 1.57, 0, 3.4, 1.64, 0, 0, 0, 0, 0\}$.

Finally, I add the micro-level routine activity features to the classification model, presented earlier in Equation 2.1. I define ten new features, $f_i(\theta_p)$ for $i = [1, 10]$ quantify the importance of venue i in people's routes:

$$f_i(\theta_p) = \sum_{r \in R(\theta_p)} VF\text{-}IRF(c_i, r) \tag{3.1}$$

where R is a function that returns the routes of all users which end up within 200-meters of point p, and $\theta_p = \{p\}$ (or $\theta_p = \{p, t_1, t_2\}$ in case of building temporal models). I define an additional feature that quantifies the density of routes at point p:

$$f(\theta_p) = \sum_{r \in R(\theta_p)} |r| \tag{3.2}$$

where $|r|$ is the size of route r (the number of venues visited by the user).

3.2.3 Evaluation

Since movement patterns varies over the course of the day (e.g., morning vs. evening movement patterns), I choose to add the micro-level daily movements features against a model with KDE and temporal features, and evaluate both crime prediction models (with and without the added features) using real crime data for 17 crime types from Chicago between 2013-07-28 and 2014-04-14. I obtained historical crime data from the City of Chicago data portal.[2] These data are extracted from the Chicago Police Department's Citizen Law Enforcement Analysis and Reporting (CLEAR) system. This system includes up-to-date criminal records representing the current judgement of the criminal justice system. For example, if a crime is recorded within CLEAR, but subsequent investigation determines that no crime actually occurred, then the record will be removed. The data included 196,347 incidents occurring between 2013-07-28 and 2014-04-14. I extracted the latitude-longitude location at block-level resolution, the timestamp, and the type of each crime incident. The data covered 17 crime types, which are shown in Table 3.2 along with per-type frequencies. I built a grid of 200-meters squares, and I set the size of the training window to 7 days and the prediction window to the day following the training window. Then, I repeatedly trained and predicted, moving the model ahead by 1 day each iteration. This setup emulates the use of my model in a practical situation where I have historical crime records and I want to predict crimes for the following day. For each crime type T, I ran 254 predictions covering 2013-08-04 and 2014-04-14. I compared my venue-enhanced model with a baseline containing only the KDE and temporal features. I evaluated both models using surveillance plots (presented in Section 2.4). Since I ran my prediction models on multiple testing days, I required a method for aggregating the daily surveillance plots and estimating the overall performance of the models. Gerber proposed a micro-level aggregation method for surveillance plots [10]. In this approach, one sums the number of true crimes of type T occurring within the $x\%$ most

[2]City of Chicago Data Portal: https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present/ijzp-q8t2

Crime Type	Frequency(%)
THEFT	46,767 (23.82%)
BATTERY	34,509 (17.58%)
NARCOTICS	22,209 (11.31%)
CRIMINAL DAMAGE	19,588 (9.98%)
OTHER OFFENSE	11,731 (5.97%)
BURGLARY	11,642 (5.93%)
ASSAULT	11,357 (5.78%)
DECEPTIVE PRACTICE	9,278 (4.73%)
MOTOR VEHICLE THEFT	7,643 (3.89%)
ROBBERY	7,602 (3.87%)
CRIMINAL TRESPASS	5,441 (2.77%)
WEAPONS VIOLATION	1,987 (1.01%)
PUBLIC PEACE VIOLATION	1,886 (0.96%)
OFFENSE INVOLVING CHILDREN	1,555 (0.79%)
SEX OFFENSE	1,379 (0.70%)
PROSTITUTION	937 (0.48%)
INTERFERENCE WITH PUBLIC OFFICER	836 (0.43%)
Total	196,347

Table 3.2: Frequency of historical crime records in Chicago.

threatened area for each day, according to the model's predictions. For example, assume that I want to aggregate the surveillance plots for two testing days d_1 and d_2. Also assume the following: (1) the prediction area contains 100 square prediction cells; (2) the prediction for d_1 ranks cell c_{41} as most threatened, and the prediction for d_2 ranks cell c_{15} as most threatened; (3) 50 actual crimes occurred in c_{41} on day d_1 and 20 actual crimes occurred in c_{15} on d_2; and (4) there were 100 actual crimes on day d_1 across the city and there were 25 actual crimes on day d_2 across the city. The resulting y-value for $x = 1\%$ in the aggregated surveillance plot is calculated as $\frac{50+20}{100+25}$. In words, this y-value is the fraction of crime occurring in the aggregated, most-threatened 1% of the area as predicted by the model. Each aggregated plot contains three series: "KDE+T", "+R", and "Δ-Peak @(x,y)". The first series, "KDE+T", represents the aggregated performance of 254 predictions using only historical crime density and temporal features. The second series, "+R", reflects the aggregated performance after adding the routine activity (route) features to the KDE+T model. Finally, the Δ series

shows the absolute difference along the y-axis between the two models. The x-y location of peak difference is indicated with "Peak @ (x,y)". In Table 3.3, I report the area under curve (AUC) for the crime prediction models along with the peak gain. Note, not all of the surveillance areas (the x-axis) are equally important. Public safety officials have limited resources (e.g., patrols) and cannot intervene across the entire study area. Therefore, peak gains that are closer to $x = 0$ are more operationally relevant. For example, both PUBLIC PEACE VIOLATION and WEAPONS VIOLATION, shown in Figure 3.4, have peak gain values of 0.09 (y-axis) that are achieved by surveilling the predicted 25% and 40% most threatened areas respectively. Therefore, the gain due to routine activity features for the former crime type is more operationally relevant because it can be achieved with the use of fewer police resources (25% patrol coverage versus 40% patrol coverage).

Given the high frequency of some crime types, even small peak gains along the y-axis can result in significant social benefit. For example, across my 254-day prediction window there were 19,588 incidents of type CRIMINAL DAMAGE. A peak gain of 3% along the y-axis (see Table 3.3) would result in the capture of 587 additional incidents using the same resource expenditure (a patrol coverage of 64%). The average peak gain in incident capture across all 17 crime types is 8% when comparing the baseline model with the activity-enhanced model.

Crime Type	KDE+T (AUC)	+R (AUC)	Δ-Peak
THEFT	0.65	**0.68**	0.06@54%
BATTERY	0.71	**0.74**	0.04@43%
NARCOTICS	0.76	**0.83**	0.18@25%
CRIMINAL DAMAGE	0.65	**0.67**	0.03@64%
OTHER OFFENSE	0.64	**0.69**	0.08@50%
BURGLARY	0.68	**0.70**	0.07@51%
ASSAULT	0.67	**0.71**	0.07@51%
DECEPTIVE PRACTICE	0.69	**0.75**	0.16@25%
MOTOR VEHICLE THEFT	0.65	**0.68**	0.06@54%
ROBBERY	0.69	**0.73**	0.07@54%
CRIMINAL TRESPASS	0.65	**0.72**	0.14@25%
WEAPONS VIOLATION	0.65	**0.68**	0.09@40%
PUBLIC PEACE VIOLATION	0.65	**0.67**	0.09@25%
OFFENSE INVOLVING CHILDREN	0.58	0.58	0.04@50%
SEX OFFENSE	**0.59**	0.58	0.02@02%
PROSTITUTION	0.66	**0.68**	0.13@25%
INTERFERENCE WITH PUBLIC OFFICER	**0.62**	0.61	0.04@25%

Table 3.3: Area under curve performance of Density + Temporal models (KDE+T) and Density + Temporal + Routines models (+R) along with the peak surveillance plot gains from adding the routines (Δ-Peak).

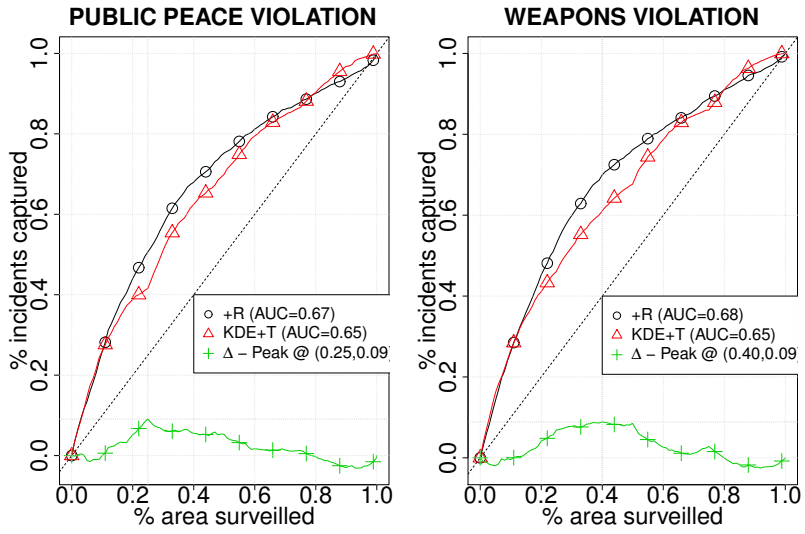


Figure 3.4: Surveillance plots using three series: Density+Temporal (KDE+T), Density+Temporal+Routines (+R), and the gains from adding the routines (the location of peak gain is indicated with Δ-Peak).

Crime Type	Residence	Professional & Other Places	Shop & Service	Travel & Transport	Food
NARCOTICS	1.33	0.73	3.11	0.47	0.07
DECEPTIVE PRACTICE	0.94	1.82	4.56	-0.19	3.02
CRIMINAL TRESPASS	0.69	0.96	2.06	1.17	1.18
PROSTITUTION	**6.92**	-5.61	-3.53	1.43	-3.78
PUBLIC PEACE VIOLATION	-0.25	-2.34	-0.14	-0.01	-1.57
WEAPONS VIOLATION	-0.08	-0.25	-1.10	-0.20	0.52
OTHER OFFENSE	2.26	0.37	2.07	6.23	-0.32
ASSAULT	2.17	2.17	3.23	-0.44	0.82
ROBBERY	2.61	0.69	4.13	-2.51	1.11
BURGLARY	3.16	-0.96	1.44	-3.43	0.14
MOTOR VEHICLE THEFT	2.72	0.13	2.56	0.16	-0.63
THEFT	1.44	-1.64	1.94	0.35	-0.47
OFFENSE INVOLVING CHILDREN	-1.35	-0.76	0.59	0.05	-1.38
INTERFERENCE WITH PUBLIC OFFICER	3.59	-1.74	-5.82	2.58	0.33
BATTERY	2.49	2.06	1.69	-1.29	0.80
CRIMINAL DAMAGE	2.90	0.41	2.20	-1.76	0.56
SEX OFFENSE	-0.71	-0.04	1.06	-0.12	0.58

Table 3.4: Averaged coefficients for the travel routine variables for 254 days of predictions with respect to the 17 crime types.

Crime Type	Nightlife Spot	Outdoors & Recreation	Arts & Entertainment	College & University	Event
NARCOTICS	-1.88	-2.00	-3.75	-4.09	-3.23
DECEPTIVE PRACTICE	0.94	-1.50	-6.01	-3.57	-1.93
CRIMINAL TRESPASS	-0.63	-1.44	-2.58	-1.85	-2.27
PROSTITUTION	**7.54**	**8.87**	-3.19	**1.89**	4.79
PUBLIC PEACE VIOLATION	1.26	2.12	-0.37	**1.56**	2.25
WEAPONS VIOLATION	1.11	0.27	-0.57	**0.79**	1.37
OTHER OFFENSE	-1.22	-1.99	-4.16	-4.54	-2.80
ASSAULT	-0.67	-2.06	-2.07	-3.52	-2.37
ROBBERY	-0.60	-1.70	-0.06	-3.09	-3.71
BURGLARY	1.76	-0.54	2.70	-4.89	-1.08
MOTOR VEHICLE THEFT	0.77	-0.10	1.64	-6.76	-2.48
THEFT	-1.25	-1.43	-3.08	-3.64	-1.76
OFFENSE INVOLVING CHILDREN	1.68	0.48	-0.62	0.12	0.73
INTERFERENCE WITH PUBLIC OFFICER	3.16	3.54	-5.70	-1.66	1.97
BATTERY	0.85	-1.62	-1.92	-5.11	-1.99
CRIMINAL DAMAGE	0.30	-0.65	-0.58	-3.74	-2.02
SEX OFFENSE	-0.92	-1.65	0.13	0.51	1.63

Table 3.5: Averaged coefficients for the travel routine variables for 254 days of predictions with respect to the 17 crime types. Interesting findings include: College & University venues positively correlate with only few crime types such as SEX OFFENSE and WEAPONS VIOLATION; Outdoors & Recreation and Nightlife Spot venues correlate with PROSTITUTION.

To better understand the correlation between crime and individuals' daily routines as inferred from social media, I inspected the fitted parameters of my model. I obtained coefficients which reflect the importance of each of the micro-level daily routine features, and I averaged the coefficients across the 254 predictions. Tables 3.4 and 3.5 show the averaged coefficients, which indicate some interesting relationships. For example, travel routines that focus on College & University venues correlate negatively with most crime types except for a few such as SEX OFFENSE and WEAPONS VIOLATION. The coefficients for PROSTITUTION show that such incidents correlate with individuals' routine travel to Outdoors & Recreation and Nightlife Spot activities. The results of my analyses, in this section, indicate (1) particular venue types and the crimes with which they are associated; (2) that visitation to these venues can be inferred from visitors' social media content; and (3) that inferences about venue visitation improve a baseline spatio-temporal predictive model.

3.2.4 Conclusions

Routine activity theory links criminal activities to individuals' daily movements and lifestyles. Researchers have used this theory to explain crime trends, primarily on the basis of macro-level survey data sets. As a first method for incorporating localized information in crime prediction models, I present a statistical approach for automatically quantifying individuals' routine activities using geolocated social media. Using these activity measurements, I have been able to improve the performance of a baseline crime prediction model that uses historical crime density and temporal features. This enhanced model improves crime prediction for 15 out of 17 crime types [44].

3.3 Area-specific crime prediction models

3.3.1 Introduction

Crime varies from neighborhood to neighborhood [27, 42]; however, current crime prediction models have ignored this variation, favoring global models (e.g., of entire cities) due to a lack of observations at finer resolutions (e.g., ZIP codes) [5–10, 43, 44]. For example, a global model might estimate the correlation between spatial density of bus stops and the occurrence and non-occurrence of thefts. Public safety officials might use a positive correlation between these variables to justify the allocation of resources (e.g., police patrols) to areas with the highest density of bus stops. However, this global correlation might not hold within each sub-area. The true correlations might be positive in one sub-area and negative in another. Area-specific models can reveal such insights, but building such models is complicated by a lack of training data at finer resolutions. Also, crime prediction methods, such as the ones mentioned in Section 2.1, estimate global models using information aggregated from many smaller areas. The models assume that the smaller areas are homogeneous with respect to crime occurrence and the various criminogenic factors studied. These models and their assumptions are at odds with evidence that crime is not homogeneous across space [27]. To address this gap, I develop and test two area-specific crime prediction models. In the first approach, I use hierarchical models [62]. In the second approach, I use regularized multi-task learning models [41]. Both approaches address the sparsity issue by sharing information across smaller areas during the learning process.

3.3.2 Mathematical approach

In this section, I present the mathematical formulation of two global models and three area-specific models.

Global model

The first baseline model is a standard global model, introduced in Section 2.2. Beside the crime density feature, I included 11 spatial features. The spatial features, shown in Table

Description	Type
Major Streets	Distance
Police Stations	Distance
Police Stations	Density
Hospitals	Distance
Hospitals	Density
Bus stops	Distance
Bus stops	Density
Parks	Distance
Parks	Density
Water ways	Distance
Pedestrian ways	Distance

Table 3.6: Spatial features included in area-specific crime prediction models. Distance features indicate linear distance from an analysis point to the spatial entities, whereas density features quantify the spatial density of those entities at that point.

3.6, were extracted from the City of Chicago data portal. Given a spatial analysis point p, I calculated the linear distance from p to each Distance feature in Table 3.6 (e.g., linear distance to a major street). I also calculated the spatial density of each Density feature at point p by yet another KDE (e.g., the value at p of the KDE built from police station locations). I choose to incorporate spatial features in the baseline global model because they are closely related, in concept, to area-specific modeling and how their values would change from one area to another.

Global model with area indicator (Global-A)

My second baseline augments the above global model with an additional categorical feature which takes on the value of the ZIP code that covers point p. I obtained Zip code boundaries from Chicago data portal, and used them to segment Chicago into its 61 existing ZIP codes and grouped the crime data accordingly (Figure 3.5).

Pooled model

My third baseline model is a pooled model that contains one independently fitted model for each of the 61 ZIP codes covering Chicago. The structure of each ZIP code-specific model is identical to the global model, but fitting is restricted to data in each area. Given infinite

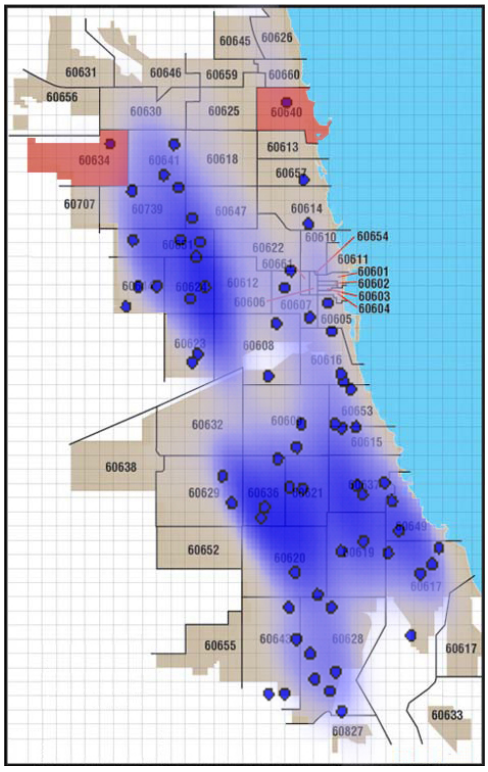


Figure 3.5: The 61 ZIP code-based modeling areas within Chicago, overlaid with a KDE of assault locations in blue. Note the sparsity of incidents in areas 60634 and 60640, which are highlighted in the top left and right, respectively.

training observations, one would expect the pooled model to outperform the global model under the hypothesis that crime is not homogeneous across spatial regions [27]. In practice, many areas have sparse data, thus complicating the direct application of pooled models. In the next two sections, I introduce area-specific models that mitigate sparsity by sharing information across different areas during the learning process.

Hierarchical model

Hierarchical models, also known as multilevel models, provide a framework for explicitly capturing the structure that may exist in categorical variables [62]. For example, one may wish to measure the performance of students in different classes within various schools, which in turn reside in districts. This structure allows model parameters to vary across levels in the

model hierarchy. In the present work, I used a varying-intercept hierarchical model given by

$$F(f_1(\theta_p), ..., f_{12}(\theta_p)) = \frac{1}{1+e^{-\left(\beta_{0[j]}+\prod_{i=1}^{12}\beta_i * f_i(\theta_p)\right)}}$$
$$\beta_{0[j]} \sim N\left(U_j, \sigma^2_{\beta_0}\right) \tag{3.3}$$

where $\beta_{0[j]}$ is the intercept of area j. Herein, the variability of the intercepts between different areas is assumed to be Gaussian with a mean U_j given by each area's data, and the within-area variance $\sigma^2_{\beta_0}$.

Multi-task model

Lastly, I implemented a regularized multi-task model [41]. This approach uses a kernel function with a task-coupling parameter to model each area as a separate task. The proposed kernel function overcomes the sparsity of area-specific models by encoding the relationship between tasks. The kernel is defined as

$$\phi\left((x, j)\right) = \left(\frac{x}{\sqrt{\mu}}, \underbrace{0, ..., 0}_{j-1}, x, \underbrace{0, ..., 0}_{J-j}\right) \tag{3.4}$$

where $j = 1, ..., J$ is the area index such that J is the total number of areas (61 in the case of Chicago), x is a vector of values $\langle f_1(\theta_p), ..., f_{12}(\theta_p)\rangle$, and μ is a trade-off parameter that balances the shared contribution of the tasks. Under this representation, each observation will fill in its values at the appropriate ZIP code location and in a shared location weighted by $\frac{1}{\sqrt{\mu}}$.

3.3.3 Evaluation

Similar to the experimental setup in Sections 2.3 and 3.2.3, I created a grid of 200-meter squares that covered Chicago. I set the training window to one month and the prediction window to the day following the training window. This design reflects a practical configuration where police officials use recent crime records to predict the next day's crime. For the pooled model, I further divided both training and testing data points into 61 subsets based on their

geo-location within the different ZIP code areas.

I evaluated models in the following way: (1) I chose 70 random testing days between 2013-08-28 and 2014-04-14; and (2) I trained Global, Global-A, and Pooled models using LibLinear's L2-regularized logistic regression solver (L2R_LR) [26]; multi-task models using LibLinear's L2-regularized SVM (L2R_L2LOSS_SVC) [26]; and hierarchical models using Glmer from the Lme4 package in R. Table 3.7 shows the aggregated AUC of the surveillance plots from the 70 prediction days. Next, I compared the performance of the Global, Global-A and Pooled models in order to select a single baseline for comparison with my two area-specific models. Table 3.7 shows that Pooled models were outperformed by Global and Global-A on all crime types, Global outperformed Global-A on 8 crime types, and Global-A outperformed Global on the remaining 9 crime types. However, I noticed that in most of these 9 crime types, the gain of Global-A over Global is not significant. To further explore the difference between these two models, I investigated the peak gains in the surveillance plots (points of maximal difference between the models). I found that the Global model had higher peak gains than the Global-A model in 10 out of 17 crime types. As a result, I selected the Global model as my baseline in the remaining analyses. My area-specific models exhibit improved performance over the Global model for 12 out of 17 crime types (see Table 3.7).

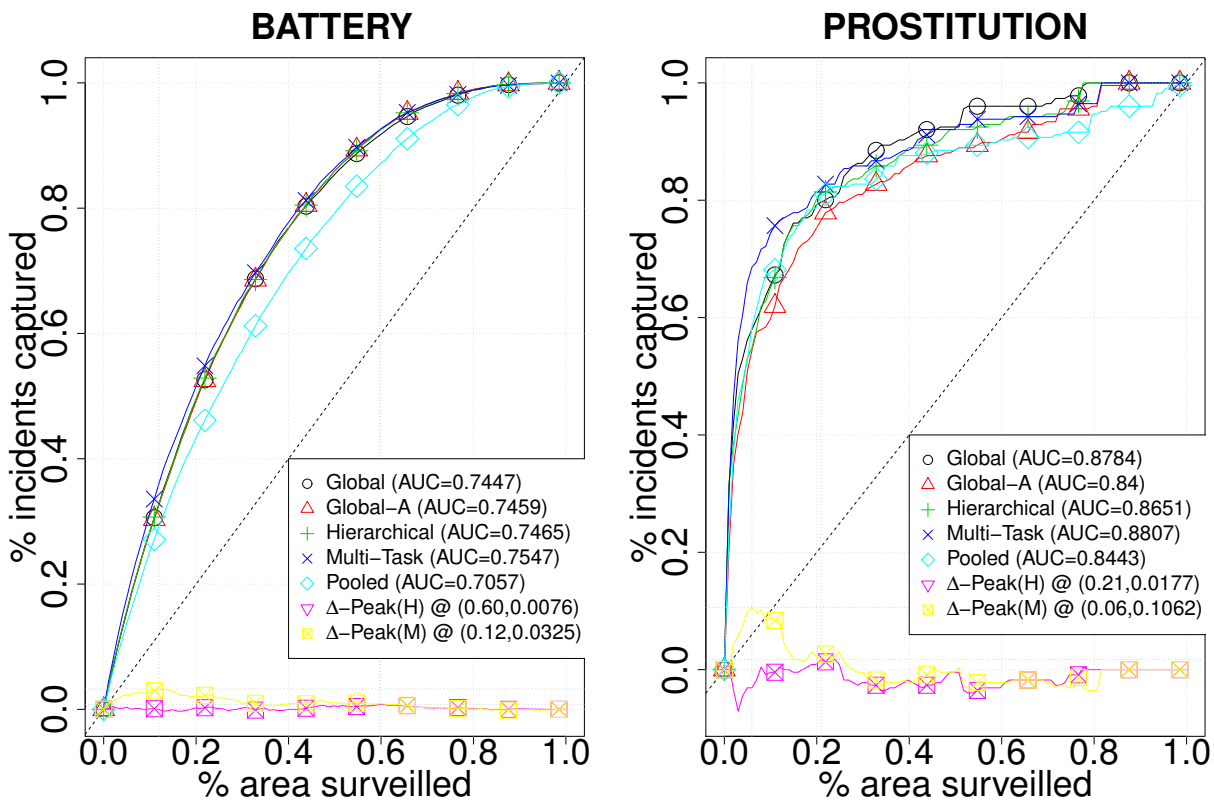


Figure 3.6: Aggregated surveillance plots of battery and prostitution crimes. Gains from using the hierarchical model are calculated as Hierarchical - Global, and this gain peaks at the location indicated by $\Delta - Peak(H)$. The gains from using the multi-task model are calculated as Multi-Task - Global, and this gain peaks at the location indicated by $\Delta - Peak(M)$.

Crime Type	Global	Global-A	Pooled	Hierarchical	Multi-Task
THEFT	0.7433	0.7500	0.7088	0.7513	**0.7603**
BATTERY	0.7447	0.7459	0.7057	0.7465	**0.7547**
NARCOTICS	0.8402	**0.8484**	0.7846	0.8482	0.8469
CRIMINAL DAMAGE	0.6886	0.6935	0.6515	0.6954	**0.6989**
OTHER OFFENSE	0.7076	0.7088	0.6678	0.7102	**0.7138**
BURGLARY	0.7031	0.7092	0.6768	0.7098	**0.7157**
ASSAULT	0.7382	0.7386	0.6901	0.7405	**0.7439**
DECEPTIVE PRACTICE	0.7723	0.7752	0.7269	0.7771	**0.7796**
MOTOR VEHICLE THEFT	0.7023	0.6992	0.6520	0.7022	**0.7051**
ROBBERY	0.7786	0.7747	0.7395	0.7810	**0.7832**
CRIMINAL TRESPASS	0.7874	0.7893	0.7620	0.7933	**0.8050**
WEAPONS VIOLATION	**0.7853**	0.7726	0.7421	0.7843	0.7744
PUBLIC PEACE VIOLATION	**0.7719**	0.7461	0.7121	0.7675	0.7614
OFFENSE INVOLVING CHILDREN	**0.6997**	0.6674	0.6557	0.6934	0.6825
SEX OFFENSE	**0.7080**	0.6681	0.6016	0.7006	0.6978
PROSTITUTION	0.8784	0.8400	0.8443	0.8651	**0.8807**
INTERFERENCE WITH PUBLIC OFFICER	0.8169	0.7823	0.7621	**0.8187**	0.7756

Table 3.7: Performance on 70 random prediction days with respect to the 17 crime types.

Crime Type	Δ-Peak (H-G)		Δ-Peak (M-G)		Δ-Peak (G-GA)		Δ-Peak (G-P)	
	X	Y	X	Y	X	Y	X	Y
THEFT	0.43	0.0144	0.35	0.0394	0.01	0.0042	0.03	0.0689
BATTERY	0.60	0.0076	0.12	0.0325	0.18	0.0055	0.30	0.0785
NARCOTICS	0.12	0.0155	0.37	0.0141	0.04	0.0057	0.05	0.1926
CRIMINAL DAMAGE	0.19	0.0289	0.19	0.0360	0.54	0.0033	0.42	0.0684
OTHER OFFENSE	0.11	0.0190	0.06	0.0356	0.55	0.0130	0.39	0.0752
BURGLARY	0.28	0.0256	0.37	0.0332	0.65	0.0068	0.46	0.0506
ASSAULT	0.35	0.0111	0.12	0.0202	0.30	0.0077	0.28	0.0956
DECEPTIVE PRACTICE	0.02	0.0287	0.02	0.0353	0.14	0.0107	0.15	0.1050
MOTOR VEHICLE THEFT	0.02	0.0157	0.44	0.0141	0.35	0.0204	0.36	0.1027
ROBBERY	0.21	0.0219	0.21	0.0225	0.08	0.0209	0.13	0.0945
CRIMINAL TRESPASS	0.27	0.0189	0.12	0.0447	0.23	0.0161	0.15	0.0692
WEAPONS VIOLATION	0.65	0.0174	0.56	0.0282	0.16	0.0521	0.15	0.1323
PUBLIC PEACE VIOLATION	0.19	0.0216	0.11	0.0275	0.13	0.0589	0.32	0.1139
OFFENSE INVOLVING CHILDREN	0.04	0.0167	0.16	0.0335	0.37	0.0766	0.52	0.0885
SEX OFFENSE	0.65	0.0172	0.18	0.0172	0.36	0.1043	0.31	0.2546
PROSTITUTION	0.21	0.0177	0.06	0.1062	0.01	0.1062	0.79	0.0841
INTERFERENCE WITH PUBLIC OFFICER	0.07	0.0534	0.05	0.0243	0.45	0.0777	0.09	0.1505

Table 3.8: Peak gains in the aggregate surveillance plot's y-value when using Hierarchical (H) and Multi-task (M) models instead of the global (G) model. We chose to include the peak gains over the Global model instead of the Pooled (P) and Global-A (GA) models since the latter were outperformed by other models for most crime types. The x-values indicate percentage area surveilled according to model predictions, and the y-values indicate gains achieved by surveilling this x% according to the H or M models instead of the G model. In words, the first row indicates (1) that, when surveilling 43% of the city, one can capture 1.4% more crime by prioritizing according to H rather than G, and (2) that, when surveilling 35% of the city, one can capture 3.9% more crime by prioritizing according to M rather than G.

In practice, police departments cannot cover more than a small fraction of the city, so I emphasize gains over global models that are achieved for small values on the x-axis of my aggregate surveillance plots. For example, Table 3.7 indicates that the multi-task model gains 0.01 over the global model for battery and 0.0023 over the global model for prostitution. This might suggest that the gains were larger for battery; however, the gains are calculated assuming 100% surveillance of the study region, which is infeasible given limited resources. If I instead inspect the x-axis location of peak gains achieved by the multi-task model over the global model (Figure 3.6), I see that, for battery, peak gain of 3.25% is achieved at 12% surveillance versus peak gain of 10.6% being achieved at 6% surveillance for prostitution. Thus, multi-task modeling produces greater returns on investment for prostitution versus battery (assuming uniform costs across crime types). Table 3.8 shows the peak gain for hierarchical and multi-task models over the Global baseline: 10 out of 17 and 11 out of 17 crime types have their peak gains within 25% surveillance using hierarchical and multi-task models respectively. Table 3.8 also shows the peak gain for the Global model over the Global-A and Pooled models. The latter two baselines offer straightforward solutions for area-specific modeling; however, because of crime sparseness, both area-specific baselines failed to outperform the simple Global baseline. This emphasizes the robustness of my area-specific hierarchical and multi-task models against sparse observations.

To better understand the impact of using area-specific models, I computed the number of crimes that could be targeted when patrolling according the various models (Tables 3.9 and 3.10). For example, when patrolling 5% of the city following the multi-task model instead of the Global model, police would have been in the vicinity of 94 additional thefts and 224 additional batteries. The improved patrol focus could have increased deterrence and decreased response time for undeterred crimes. It is also important to consider the potential social impact of improvements due to area-specific modeling. As a measure of severity, I considered prison sentences for the various crime types. For example, Class-1 felonies such as sexual assaults, residential burglaries, and possession of heroin or cocaine have sentencing periods of

Crime Type	Δ-Peak (H-G) Global	Δ-Peak (H-G) Hierarchical	Δ-Peak (M-G) Global	Δ-Peak (M-G) Multi-Task
THEFT	9199	173(↑1.9%)	8388	453(↑5.4%)
BATTERY	8395	70(↑0.8%)	3001	298(↑9.9%)
NARCOTICS	3495	90(↑2.6%)	4984	80(↑1.6%)
CRIMINAL DAMAGE	1874	130(↑6.9%)	1874	173(↑9.2%)
OTHER OFFENSE	876	37(↑4.2%)	553	105(↑19.0%)
BURGLARY	1573	66(↑4.2%)	1918	90(↑4.7%)
ASSAULT	2096	27(↑1.3%)	956	60(↑6.3%)
DECEPTIVE PRACTICE	528	70(↑13.3%)	528	86(↑16.3%)
MOTOR VEHICLE THEFT	84	30(↑35.7%)	1443	22(↑1.5%)
ROBBERY	1120	34(↑3.0%)	1120	39(↑3.5%)
CRIMINAL TRESPASS	982	27(↑2.7%)	656	64(↑9.8%)
WEAPONS VIOLATION	438	8(↑1.8%)	424	11(↑2.6%)
PUBLIC PEACE VIOLATION	278	6(↑2.2%)	190	12(↑6.3%)
OFFENSE INVOLVING CHILDREN	58	4(↑6.9%)	143	14(↑9.8%)
SEX OFFENSE	318	4(↑1.3%)	124	6(↑4.8%)
PROSTITUTION	181	3(↑1.7%)	135	22(↑16.3%)
INTERFERENCE WITH PUBLIC OFFICER	81	8(↑9.9%)	65	3(↑4.6%)

Table 3.9: Number of criminal incidents captured by global, hierarchical and multi-task models at the percentage at which the peak gain is achieved. We show the gain in true counts as well as in percentages of increase or decrease.

4 to 15 years, while Class-4 felonies such as aggravated assault have sentencing periods of 1 to 3 years.[3] My area-specific models offer significant social impact through improved prediction of the most severe crimes within small surveillance areas (5% to 25% most-threatened areas versus the Global baseline, see Table 3.8).

[3]Illinois prison talk:http://www.illinoisprisontalk.org/index.php?topic=23719.0

Crime Type	5%			10%		
	Global	Hierarchical	Multi-Task	Global	Hierarchical	Multi-Task
THEFT	3112	-11 (↓ 0.4%)	**94 (↑ 3.0%)**	4277	77 (↑1.8%)	**224(↑ 5.2%)**
BATTERY	1682	0 (↑ 0.0%)	**224 (↑13.3%)**	2803	9 (↑0.3%)	**272(↑ 9.7%)**
NARCOTICS	2655	**21 (↑ 0.8%)**	-22 (↓ 0.8%)	3383	**77 (↑2.3%)**	39(↑ 1.2%)
CRIMINAL DAMAGE	653	**65 (↑10.0%)**	64 (↑ 9.8%)	1135	58 (↑5.1%)	**110(↑ 9.7%)**
OTHER OFFENSE	481	27 (↑ 5.6%)	**107 (↑22.2%)**	809	57 (↑7.0%)	**82(↑10.1%)**
BURGLARY	447	40 (↑ 8.9%)	**42 (↑ 9.4%)**	737	39 (↑5.3%)	**64(↑ 8.7%)**
ASSAULT	561	-24 (↓ 4.3%)	**-1 (↓ 0.2%)**	892	13 (↑1.5%)	**54(↑ 6.1%)**
DECEPTIVE PRACTICE	853	16 (↑ 1.9%)	**34 (↑ 4.0%)**	1096	16 (↑1.5%)	**18(↑ 1.6%)**
MOTOR VEHICLE THEFT	268	**-4 (↓ 1.5%)**	-9 (↓ 3.4%)	438	**19 (↑4.3%)**	0(↑ 0.0%)
ROBBERY	465	-3 (↓ 0.6%)	**20 (↑ 4.3%)**	728	21 (↑2.9%)	**32(↑ 4.4%)**
CRIMINAL TRESPASS	443	13 (↑ 2.9%)	**60 (↑13.5%)**	629	21 (↑3.3%)	**53(↑ 8.4%)**
WEAPONS VIOLATION	106	**-3 (↓ 2.8%)**	-14 (↓13.2%)	180	**-6 (↓-3.3%)**	-15(↓-8.3%)
PUBLIC PEACE VIOLATION	109	-9 (↓ 8.3%)	**5 (↑ 4.6%)**	174	-5 (↓-2.9%)	**14(↑ 8.0%)**
OFFENSE INVOLVING CHILDREN	69	-2 (↓ 2.9%)	**0 (↑ 0.0%)**	109	**6 (↑5.5%)**	1(↑ 0.9%)
SEX OFFENSE	47	0 (↑ 0.0%)	0 (↑ 0.0%)	85	-7 (↓-8.2%)	**0(↑ 0.0%)**
PROSTITUTION	131	-7 (↓ 5.3%)	**24 (↑18.3%)**	152	-1 (↓-0.7%)	**19(↑12.5%)**
INTERFERENCE WITH PUBLIC OFFICER	101	**3 (↑ 3.0%)**	-6 (↓ 5.9%)	152	**1 (↑0.7%)**	-22(↓-14.5%)

Table 3.10: Number of criminal incidents captured by global, hierarchical and multi-task models at different surveillance levels: 5% and 10%. We show the gain in true counts as well as in percentages of increase or decrease.

3.3.4 Conclusions

The sparsity of crime in many areas complicates the application of area-specific predictive modeling. Researchers have favored global models (e.g., of entire cities) due to a lack of observations at finer resolutions (e.g., ZIP codes). These global models and their assumptions are at odds with evidence that the relationship between crime and criminogenic factors is not homogeneous across space. In response to this gap, I presented area-specific crime prediction models based on hierarchical and multi-task statistical learning. My models mitigated sparseness by sharing information across ZIP codes, yet they retain the advantages of localized models in addressing non-homogeneous crime patterns. Out-of-sample testing on real crime data indicates predictive advantages over multiple state-of-the-art global models. Using real crime records from Chicago, Illinois, USA, I tested the hypothesis that my area-specific models will outperform global models. My experiments supported my hypothesis: The area-specific models achieve better prediction performance on 12 out of 17 crime types, and I observed gains in many surveillance-feasible regions of the study area [40].

3.4 Localized kernel density estimation

3.4.1 Introduction

The KDE method has been widely adopted by police analysts for hotspot mapping. It is also included in many commercial and non-commercial spatial analysis packages such as ArcGIS, MapInfo, R, and CrimeStat III. Furthermore, KDEs have been integrated with spatial, temporal, and social media data to improve the performance of crime prediction models [6, 10, 44]. Researchers have studied KDEs and compared them to other hotspot mapping methods [18, 63, 64]. Chainey et al. argue that KDEs offer the best prediction performance across all hotspot mapping methods [18]. On the other hand, Pezzuchi and Levine argue that KDEs are not universally the best considering different crime types and environment characteristics [63, 64]. One of the reasons for this ongoing debate is that KDE performance can vary depending on how parameters are set within the estimator. There are three user-defined parameters that can affect KDE performance: grid cell size, interpolation method (kernel function), and search radius (bandwidth). Chainey has examined the effect of different grid cell sizes and bandwidths on KDE performance [65], and he concluded that varying the grid cell size has little to no impact on performance. More important is the choice of bandwidth. These findings were later confirmed by Hart and Zandbergen [66]. They also studied the effect of various interpolation methods on performance and found that among the three parameters, interpolation methods have the biggest performance impact. Researchers have offered guidance on setting the grid cell size and bandwidth [15, 21, 22, 24]. However, to date, there has been no guidance about which interpolation method should be used with respect to various scenarios. To fill this gap, I propose a novel method for generating hotspot maps using localized kernel density estimation (LKDE), where the LKDE parameters are automatically optimized using a genetic algorithm. The proposed method is motivated by ideas from image processing, specifically convolution filtering. Moreover, I examine the effect of data sparseness on the performance of the proposed method and the traditional KDE

method. This has not been studied in previous research.

3.4.2 Mathematical approach

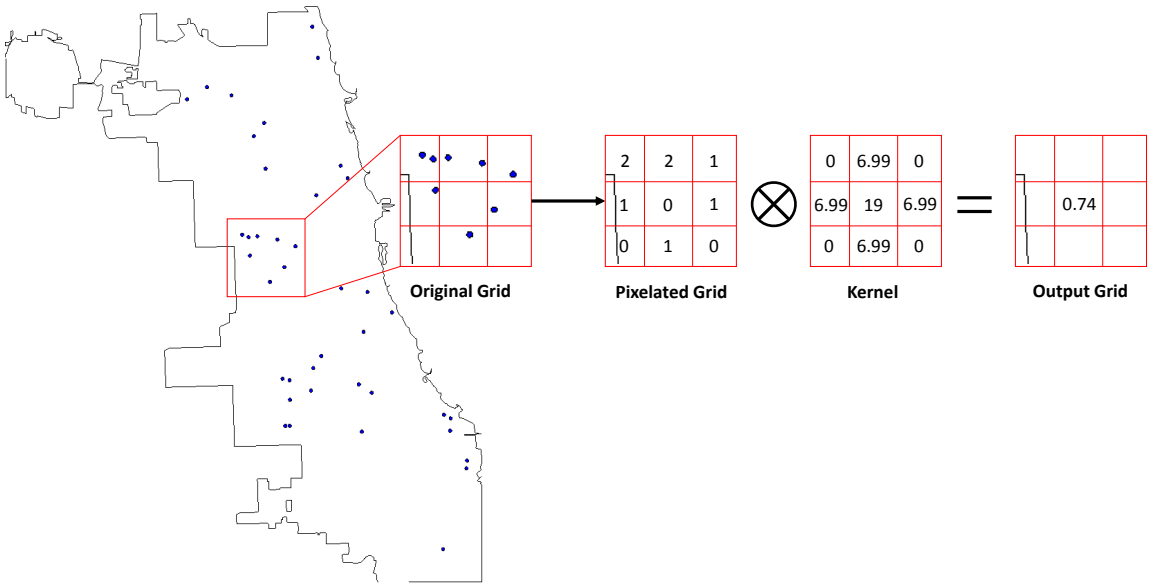


Figure 3.7: An illustration of my localized kernel density estimation approach using convolution filtering. In this illustration, I show the map of Chicago, Illinois with assault incidents shown as blue points. The LKDE process involves (1) building an overlay grid, (2) counting the frequency of incidents within each grid cell, (3) fixing the center of a convolution kernel at each cell, and (4) performing a convolutional operation. The final output represents the density estimate and will be later used to build hotspot maps. Note that zero-padding is required for grid cells on the boundary.

Inspired by concepts from image processing, specifically convolution filtering, I propose an alternative method for performing KDE that significantly reduces computational complexity, facilitates learning of a bandwidth and a non-standard kernel function, and improves the resulting estimation performance for crime prediction. Convolution filtering can be used to perform a number of basic operations such as edge detection, corner detection, and sharpening [67], and the same concepts have been used as part of convolutional neural networks [68]. In convolution filtering, we define a kernel (also referred to as convolution matrix or mask), and we apply this kernel at each pixel of the image such that the new pixel value equals a weighted average of surrounding pixel values. In the context of my problem, one can think of the study region as an image with each grid cell being a pixel and each pixel value being the

number of crime incidents in each cell. Thus, a blurring convolutional operation would result in a localized density estimate from the surrounding cells.

Figure 3.7 illustrates my localized kernel density estimation (LKDE) approach. In comparison with the traditional KDE, the size of the convolution kernel is similar to the concept of bandwidth while the weighted average is conceptually similar to the interpolation method. However, in the LKDE, the interpolation can vary with respect to each cell depending on the corresponding kernel weight, which produces a more flexible and non-standard interpolation. Moreover, the proposed approach significantly reduces the computational cost since its complexity depends on the kernel size rather than the number of incidents as in the KDE. Returning to the complexity analysis example from Section 2.2.1, given an 11-by-11 kernel, an LKDE would require approximately 1.9 million basic operation versus the 140 million required for a standard KDE. Figure 3.8 shows two sample hotspot maps produced using KDE and LKDE given the same training and testing data (the latter indicated with dark, filled circles).

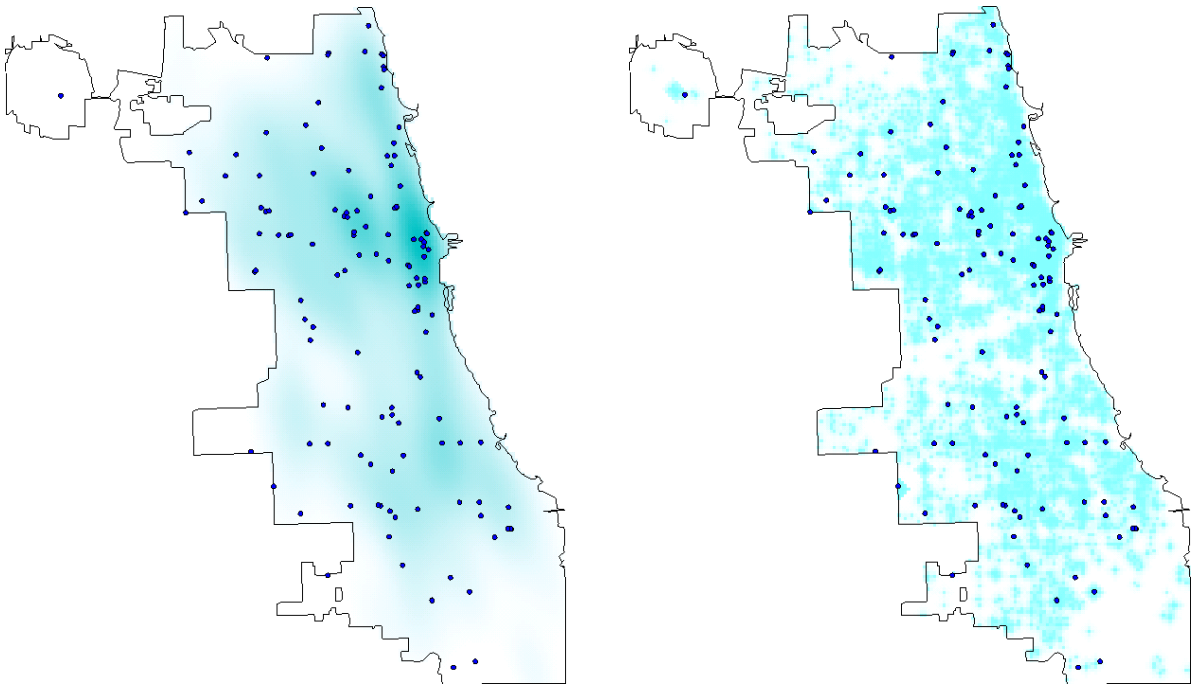
The approach described above requires decisions regarding kernel size and convolution values. In the following subsection, I present an approach for learning a dynamic-sized and dynamic-valued kernel via genetic algorithms.

Learning convolutional kernels for LKDE

The success of LKDE depends on the quality of the convolutional kernels. Since kernels can be of different sizes and shapes, and their values can span a wide range, I require a learning algorithm to identify optimal kernels. However, the solution space is large and non-convex. Therefore, I developed an evolutionary algorithm, specifically a genetic algorithm (GA), to learn the convolutional kernels. GAs require the following elements:

Encoding

Kernels can be encoded a various ways. One way is to encode each kernel value separately. For example, the 3-by-3 kernel in Figure 3.7 would be encoded as a 9-numbered chromosome. However, given the sparseness of the training data and the fact that kernel size can vary from



(a) Hotspot map produced by KDE.

(b) Hotspot map produced by LKDE.

Figure 3.8: Two hotspot maps generated by KDE and LKDE. Darker shading represents higher density estimates and predicted risk. LKDE produces a less smooth risk surface because only training points in the immediate vicinity of each cell are used for the estimation. Note that blue points are the true incident points from the testing window.

one situation to another, we should favor smaller chromosomes. Additionally, the roughly stable spatial distribution of crime suggests that the historical crime frequency of the current grid cell (i.e., the one to which I are applying the convolutional operation) should have the highest impact on the predicted risk of crime within that cell. As I move away from the current cell, historical crime frequencies should have less impact on the convolutional operation. Formally, I build exponentially decaying kernels using the following decay process:

$$N_d = N_0 e^{-\lambda d} \tag{3.5}$$

where N_0 is the value of the central grid cell (19 in the kernel of Figure 3.7), d is Manhattan distance from the central cell to the current cell, and λ is the decay rate. This setup permits the construction of dynamic-sized convolutional kernels using only three parameters: initial

value, largest considered distance ($\widetilde{d}$), and the decay rate. Each chromosome encodes these three parameters, and the optimization task is to identify the best parameterization for the observed crime distribution. Figure 3.9 shows examples of a 3-by-3 and a 5-by-5 kernel such that $N_0 = 19$, $\lambda = 1$, and $\widetilde{d}$ equals 1 and 2, respectively. Finally, I encode each chromosome as a binary vector such that each parameters is encoded using a fixed number of bits. For example, if I represent the initial value, N_0, using five bits, then the initial value can take a value between 0 and 31.

0	6.99	0
6.99	19	6.99
0	6.99	0

(a) 3x3 kernel

0	0	2.57	0	0
0	2.57	6.99	2.57	0
2.57	6.99	19	6.99	2.57
0	2.57	6.99	2.57	0
0	0	2.57	0	0

(b) 5x5 kernel

Figure 3.9: Example exponentially decaying kernels with $N_0 = 19$, $\lambda = 1$, and $\widetilde{d}$ set to 1 and 2 for Sub-Figure a and b, respectively.

Fitness function

The fitness of each chromosome is calculated as the area under the curve (AUC) of the crime surveillance plot for the prediction [10].

Selection scheme

I adopted a traditional roulette wheel selection scheme in which the probability of selecting a chromosome for the next iteration depends on its fitness value:

$$p_i = \frac{f_i}{\sum_{j=1}^{M} f_j} \tag{3.6}$$

where f_i is the fitness of chromosome i, M is the population size, and p_i is the selection probability of chromosome i.

Crossover strategy

I randomly select two chromosomes and an index. Then, the offspring are generated from the two chromosomes with values swapped at the selected index. Each chromosome will fix a subset of the bits, p, and swap the remaining subset $(1 - p)$ from the other chromosome.

Mutation strategy

I randomly choose a chromosome and an index. Then, I generate a random value between 0 and 1. Finally, in order to produce the mutated offspring, I set the bit at the chosen index to 1 if the generated number is greater than 0.5; and 0 otherwise.

Stopping condition

I used a maximum number of iterations strategy as the stopping condition. After reaching that number, I stop the GA and return the best chromosome.

3.4.3 Evaluation

In the third approach for creating localized crime prediction models, I proposed an alternative way for estimating historical crime density at a location using local historical incidents. Therefore, I choose to compare the new method against a global model with only a KDE feature. I followed the same experimental settings from Section 3.3.3: (1) I created a grid of 200-meter squares covering the official city boundary of Chicago; and (2) I enumerated 70 random testing days between 2013-08-28 and 2014-04-14 for each crime type; (3) for each testing day and crime type, I fitted KDE and LKDE models using crime data prior to the testing day; (4) I used each fitted model to predict hotspots for the associated testing day, and I aggregated all testing days for a model and crime type following the process described by Gerber [10].

For example, to evaluate the burglary KDE model on the testing day of 2013-09-29, I used all burglary incidents between 2013-8-29 and 2013-09-28. I used the `kde` function from the `ks` package in R, optimizing the bandwidth with the `Hpi` heuristic. I then computed the surveillance plot for the fitted KDE using burglaries occurring on 2013-09-29. After repeating this process for the 70 testing days I aggregated the surveillance plots to arrive at a final evaluation of the KDE model on the burglary crime type. I then repeated this process for each crime type.

Estimating the LKDE proceeded similarly to the example above, except that prior to computing the density estimate I optimized an exponentially decaying convolutional kernel

using a GA. I used the final three days of the training data for evaluating the fitness of chromosomes (kernel size, decay rate, and interpolation values), and I used one month prior to these days for training during GA optimization. For example, for the testing day 2013-09-29, I optimized the convolution kernel using a GA trained on incidents between 2013-8-26 and 2013-09-25 and evaluated the chromosomes on burglaries between 2013-09-26 and 2013-09-28. I set the population size to 50, I used 50% crossover rate, 5% mutation rate, and I set the maximum number of iterations to 30. The optimal chromosome (convolution kernel) was then used to predict burglaries for the testing day 2013-09-29. Lastly, I aggregated the resulting surveillance plots as described above to get a final surveillance plot for burglary, which was compared with the aggregated plot for KDE. Note that in order to match the experimental settings for both KDE and LKDE, I evaluated the `Hpi` function on the same data points used in the GA: one month prior to three days from the testing day.

Crime Type	Overall Performance		Δ-Peak		Average Kernel
	KDE	LKDE	X	Y	Size
THEFT	0.7156	**0.7860**	0.19	14.20%	17x17
BATTERY	0.7269	**0.7923**	0.18	13.30%	16x16
NARCOTICS	0.8331	**0.8749**	0.08	12.27%	16x16
CRIMINAL DAMAGE	0.6725	**0.7275**	0.22	10.34%	17x17
OTHER OFFENSE	0.6937	**0.7298**	0.13	7.72%	16x16
BURGLARY	0.6998	**0.7403**	0.25	8.92%	18x18
ASSAULT	0.7204	**0.7559**	0.13	8.65%	19x19
DECEPTIVE PRACTICE	0.7433	**0.7657**	0.02	8.01%	19x19
MOTOR VEHICLE THEFT	0.6987	**0.7125**	0.13	3.77%	20x20
ROBBERY	0.7463	**0.7768**	0.22	7.73%	21x21
CRIMINAL TRESPASS	0.7592	**0.7991**	0.05	15.37%	20x20
WEAPONS VIOLATION	**0.7826**	0.7764	0.11	4.98%	24x24
PUBLIC PEACE VIOLATION	**0.7401**	0.7342	0.06	6.88%	22x22
OFFENSE INVOLVING CHILDREN	**0.6837**	0.6722	0.04	4.07%	23x23
SEX OFFENSE	**0.6745**	0.6627	0.18	4.04%	35x35
PROSTITUTION	0.8458	**0.8552**	0.01	25.22%	17x17
INTERFERENCE WITH PUBLIC OFFICER	**0.7889**	0.7810	0.09	4.70%	22x22

Table 3.11: Performance on 70 random prediction days with respect to the 17 crime types along with the peak gains of using the LKDE instead of the KDE. The x-values in column four indicate area coverage and the y-values indicate gains achieved by surveilling x% according to the LKDE model instead of the KDE model.

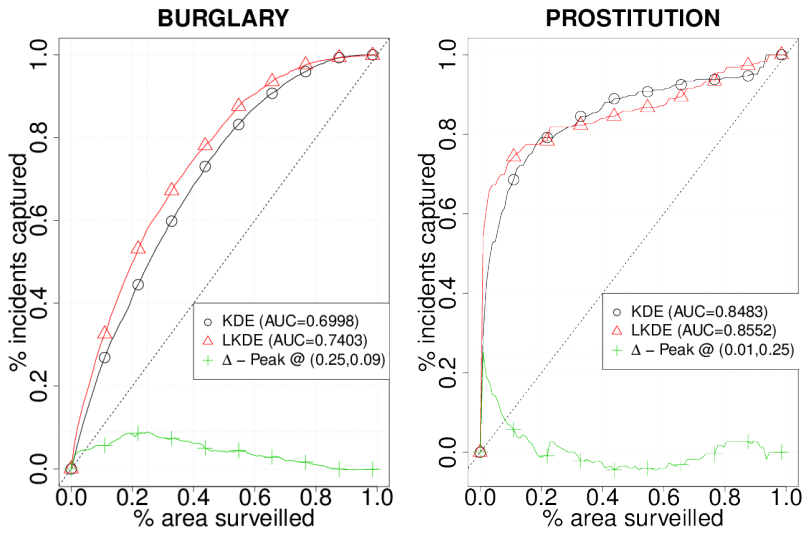


Figure 3.10: Aggregated surveillance plots of burglary and prostitution crimes. The gain series, shown in green, is computed as LKDE-KDE. The highest peak gain occurs at the location specified by $\Delta - Peak$.

Table 3.11 shows the aggregated surveillance plot AUC over 70 prediction days by crime type and model, sorted by decreasing crime frequency. LKDE outperformed KDE for 12 out of 17 crime types, often achieving substantial gains. For example, the LKDE model captures 25% more criminal incidents (absolute) than the KDE at the same 0.01 area-coverage rate. The KDE often outperformed LKDE on infrequent crime types, which constitute less than 4% of all crime. Prostitution, on the other hand, benefited from using LKDE although it has a very low frequency. This happens because such crime type repeats at the same places, and therefore, I have sufficient data to learn a good local kernel with the GA.

Given the fact that public safety officials can only cover a small percentage of the study region, I analyzed the performance gain over various area-coverage rates. For example, Table 3.11 shows that by using LKDE, we gain 4% AUC over the KDE model for burglary and 1% AUC over the KDE model for prostitution. These results might suggest that LKDE achieves greater gains for burglary than for prostitution. However, these gains are computed at 100% area-coverage, which is certainly not a feasible coverage rate in practice. If instead we examine the x-value location of peak gain achieved by the LKDE model over the KDE

model (Figure 3.10), we find that, for burglary, peak gain of 9% is achieved at 25% area coverage, versus a peak gain of 25% being achieved at 1% area coverage for prostitution. These results indicate that the practically achievable gain for LKDE is much greater for prostitution than for burglary. Table 3.11 shows the peak gain for the LKDE model over the KDE model in the fourth and fifth columns. These results indicate not only the effectiveness of LKDE in outperforming the KDE but also that these gains are often achieved for low area-coverage values (i.e., at small x-values), increasing the likelihood of practical realization.

Finally, I calculated the average kernel size learned by the GA across the 70 days (final column in Table 3.11). I observed that with sparser data (bottom rows), the optimized kernels become larger, taking more of the surrounding area into account when estimating the risk at the interpolation point. I believe that this flexibility in kernel size is a primary advantage of the LKDE method. It allows the LKDE to take advantage of dense data when possible, without affecting prediction in areas of sparse data.

3.4.4 Robustness analysis

To better understand how data sparseness affects KDE and LKDE performance, I performed a robustness analysis on theft, which is the most common crime type in the dataset. The analysis proceeded as follows: (1) I chose four random testing days; (2) for each testing day, I fixed the size of the convolutional kernel in the GA and gradually removed a percentage of random observations from the training data; (3) I learned three kernels of sizes 7x7, 9x9, and 11x11. I also performed KDE on the same data, estimating the bandwidth using the `Hpi` procedure; (4) I evaluated the KDE and LKDE models on the testing days, computing AUC scores from the aggregated surveillance plots. I then replicated these steps 10 times using different random seeds for the ablation. Figure 3.11 presents the results. I observed the following: (1) With more data, smaller kernels outperform larger ones; (2) larger kernels are more resilient to data sparseness; (3) when dealing with sparse data, I observe higher variance among different evaluations; and (4) although the KDE uses dynamic bandwidth, it is still unable to perform well on sparse data. Observations 1 and 2 corroborate my observation

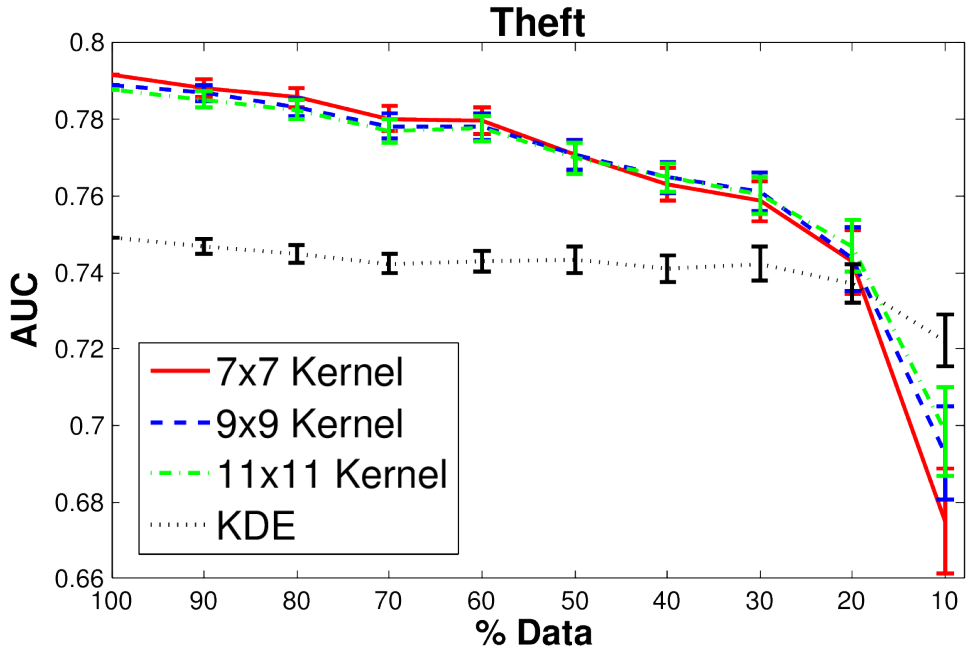


Figure 3.11: Performance of KDE and three fixed-size LKDEs on randomly chosen subsets of theft incidents.

regarding kernel sizes from Table 3.11.

3.4.5 Conclusions

I have proposed a novel approach for predicting hotpots using localized kernel density estimation optimized by a genetic algorithm. My model was inspired by concepts from image processing, and it addresses two primary limitations of traditional kernel density estimation: computational complexity and kernel function choice. My LKDE method is able to leverage dense data when available and will typically produce focused, more accurate convolution kernels in such situations. As data become sparse, the LKDE method adapts by enlarging the kernel to leverage additional data. My experiments on real crime data of 17 types from Chicago shows the effectiveness of my approach versus traditional KDE. Peak gains reach 25% (absolute) versus KDE, with many of these gains realized at small area-coverage rates. I have taken a preliminary look at robustness to sparsity, and the results support the use of adaptable convolution kernels [69].

3.5 Multi-localized kernel density estimation

3.5.1 Introduction

In section 3.4, I introduced a new method for estimating historical crime density locally using only neighboring and surrounding incidents. From the results in Table 3.11 and the robustness analysis, presented in Section 3.4.4, I observe that the performance of my LKDE model drops as data become more sparse. Table 3.11 shows that the standard KDE outperformed LKDE for 5 out of 17 crime types. These crime types constitute only 3.89% of all incidents in Chicago between 2013-07-28 and 2014-04-14 (See Table 3.2). The LKDE method address the sparsity challenge by allowing the size of the kernel to be learned automatically from data such that with fewer observations, the algorithm optimizes a larger kernel. However, after reaching a certain sparsity threshold (e.g., 10% of theft incidents in the case study in Figure 3.11), learning a bigger kernel no longer helps in achieving a better performance.

To better understand the reason behind the performance drop, we need to review the settings for optimizing a localized kernel. In summary, the settings include: 1) I used the final three days of the training data for evaluating the kernels, and 2) I used one month prior to these days for training during the optimization. For example, for the testing day 2013-09-29, localized kernels will be optimized on incidents between 2013-8-26 and 2013-09-25 and evaluated on the ones between 2013-09-26 and 2013-09-28. The assumption herein is that most recent incidents (i.e., incidents from the three days prior to the testing day) will have a similar distribution to future incidents occurring on the testing day. Therefore, the assumption is that by optimizing a good kernel for those incidents, I hope to perform well on future incidents that would follow the same distribution. I hypothesis that my distributional assumption holds true for dense crime types since within three days, I would have enough observations to estimate a crime distribution that would be representative of future incidents. On the other hand, sparser crime types would have higher distribution variance across

consecutive days, and therefore, I would have less chance in capturing the distribution of future incidents.

3.5.2 Analyzing crime distribution variance

Validating the distributional assumption would require analyzing the variance in crime distribution across different days and different crime types. In order to achieve this, I need to define a metric to measure the distance between crime patterns in two different time intervals. In this section, I propose a new metric for calculating that difference.

Following the same analogy from Section 3.4.2, I represent crimes occurring in an area of interest over a certain period of time as an image such that grid cells correspond to pixels, and the number of incidents at each cell corresponds to pixel values. Then, the problem of calculating distribution difference across different time intervals boils down to calculating the distance between images generated from those intervals. Straightforward distance metrics for comparing images include sum of squared pixel-wise differences (SSD) and sum of absolute pixel-wise differences (SAD). However, both metrics don't account for spatial relationships between pixels which are very crucial in the context of crime incidents. For example, let's have three intervals with different crime distributions (as shown in Figure 3.12) such that four incidents have occurred within each interval. Only three out four incidents have re-occurred at the same locations, and the distance measurement would only reflect the fourth incident. Using SAD for example, the distances between interval 1 and both intervals 2 and 3 are equal (i.e., $SAD(I_1 - I_2) = SAD(I_1 - I_3) = |1 - 0| + |0 - 1| + |2 - 2| + |1 - 1| = 2$). However, Figure 3.12 clearly shows that the distance between I_1 and I_2 should be different from the distance between I_1 and I_3.

I propose a new distance metric that preserve the spatial relationships between incidents by taking into account the row and column indices of each incident. I define the row-column

0	0	0	0	0	0	0
0	1	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	2	0	0
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	0	0	0	0	0	0

(a) Interval 1 (I_1)

0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	2	0	0
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	0	0	0	0	0	0

(b) interval 2 (I_2)

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	2	0	0
0	0	0	0	0	1	0
0	0	1	0	0	0	0
0	0	0	0	0	0	0

(c) Interval 3 (I_3)

Figure 3.12: Example of crime distribution of three different intervals represented as 7x7 images. This example illustrates the advantage of my row-column different approach over the traditional pixel-wise different approach such that the latter does not account for spatial relations between pixels.

difference (RCD) metric as

$$RCD(I_i, I_j) = \left| \sum_{p \in I_i} R(p) - \sum_{p \in I_j} R(p) \right| + \left| \sum_{p \in I_i} C(p) - \sum_{p \in I_j} C(p) \right| \tag{3.7}$$

where p is an incident point, $R(p)$ is the one-based row index of incident p, and $C(p)$ is the one-based column index of incident p. As incidents reoccur at different grid cells, their row and column indices affects the distance measurement while preserving the spatial relationships. Using the RCD to measure the distances in the example in Figure 3.12:

$\text{RCD}(I_1 - I_2) = |(2+4+4+6) - (2+4+4+6)| + |(2+3+5+5) - (3+3+5+5)| = 1$

$\text{RCD}(I_1 - I_3) = |(2+4+4+6) - (5+4+4+6)| + |(2+3+5+5) - (6+3+5+5)| = 7$

Next, I validated the distributional assumption using the following steps: I (1) obtained daily criminal incidents between 2013-01-01 and 2013-12-31, (2) generated 365 images from the obtained incidents, (3) calculated the pair-wise RCD differences between the 365 images, (4) filtered outliers, i.e., removed any images with whose distance is less than $q_1 - 1.5 * iqr$ or greater than $q_3 + 1.5 * iqr$ where q_1 is the first quartile, q_3 is the third quartile, and iqr is the interquartile range, and (5) computed the variance of RCD differences between each image and other images. Figure 3.13 shows the variance of RCD differences for nine exemplary crime types. As the frequency of a crime type decreases, I observed an increasing trend in the

mean variance. Also, infrequent crime types tend to have a wider range than denser crime types. This indicates that with infrequent crime types, not only crime distributions change from day to day but also some days will have a drastically different distribution such that the variance for those days is twice as high as the average variance. Table 3.12 shows the complete statistical summary for all 17 crime types from Table 3.2.

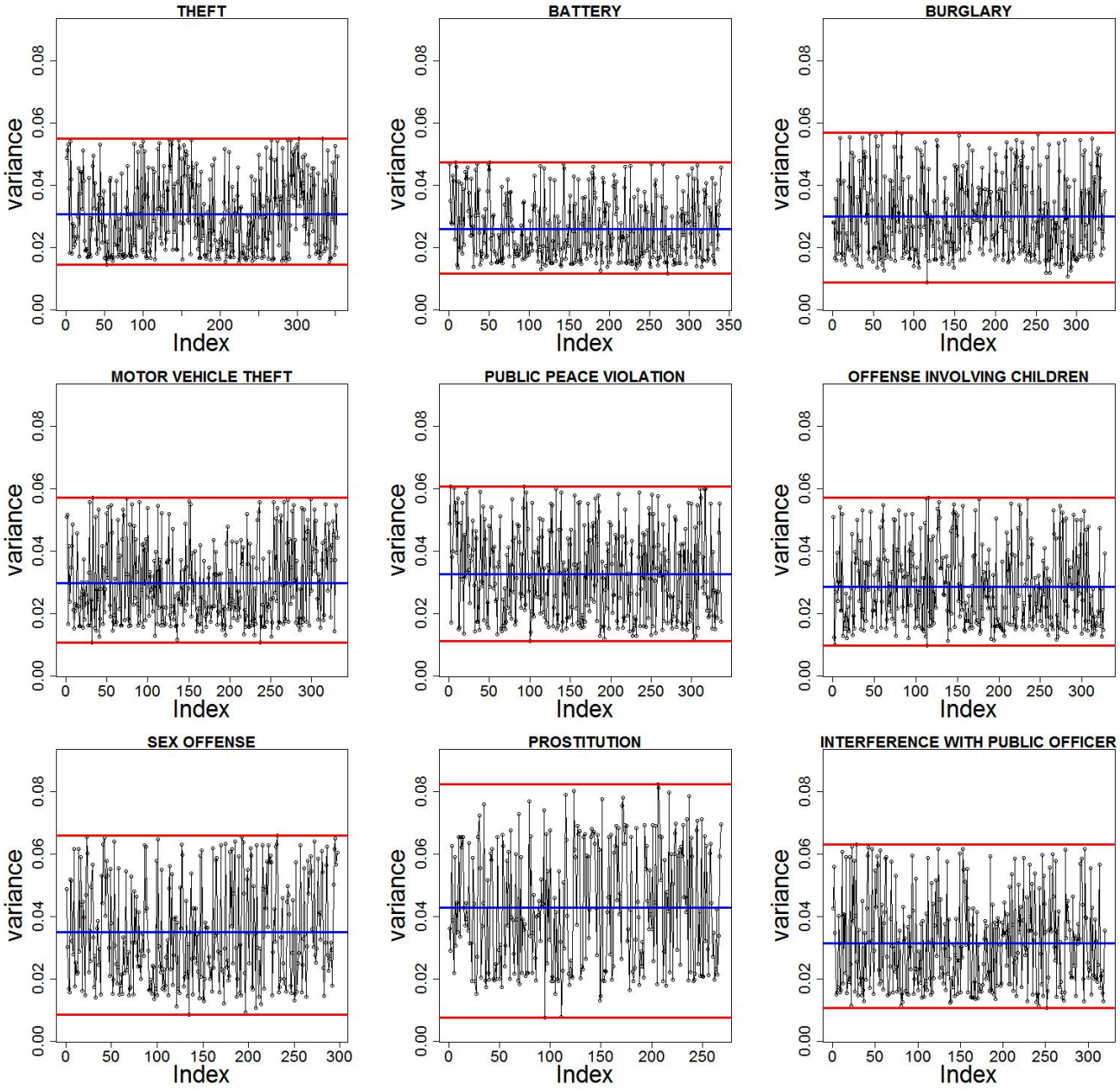


Figure 3.13: Variance of RCD differences for daily crime distributions in 2013 for nine crime types. The blue line is the mean variance while the red lines are the minimum and maximum variances. Less frequent crime types tend to a have higher variance mean and a wider range.

Crime Type	Min	Q1	Median	Mean	Q3	Max	Range
THEFT	0.0144	0.0187	0.0281	0.0306	0.0414	0.0549	0.0226
BATTERY	0.0116	0.0171	0.0231	0.0260	0.0335	0.0474	0.0164
NARCOTICS	0.0113	0.0165	0.0236	0.0266	0.0350	0.0499	0.0184
CRIMINAL DAMAGE	0.0127	0.0188	0.0270	0.0302	0.0400	0.0562	0.0211
OTHER OFFENSE	0.0097	0.0165	0.0245	0.0273	0.0368	0.0528	0.0203
BURGLARY	0.0088	0.0182	0.0269	0.0300	0.0403	0.0568	0.0221
ASSAULT	0.0126	0.0186	0.0260	0.0298	0.0402	0.0558	0.0215
DECEPTIVE PRACTICE	0.0101	0.0172	0.0250	0.0282	0.0385	0.0523	0.0213
MOTOR VEHICLE THEFT	0.0106	0.0181	0.0253	0.0298	0.0401	0.0571	0.0219
ROBBERY	0.0101	0.0186	0.0271	0.0312	0.0420	0.0597	0.0234
CRIMINAL TRESPASS	0.0119	0.0176	0.0257	0.0291	0.0379	0.0561	0.0203
WEAPONS VIOLATION	0.0108	0.0184	0.0273	0.0307	0.0408	0.0598	0.0224
PUBLIC PEACE VIOLATION	0.0111	0.0188	0.0314	0.0327	0.0441	0.0606	0.0253
OFFENSE INVOLVING CHILDREN	0.0098	0.0167	0.0256	0.0286	0.0380	0.0570	0.0212
SEX OFFENSE	0.0087	0.0205	0.0304	0.0349	0.0498	0.0660	0.0293
PROSTITUTION	0.0077	0.0230	0.0405	0.0429	0.0624	0.0823	0.0393
INTERFERENCE WITH PUBLIC OFFICER	0.0106	0.0178	0.0302	0.0313	0.0407	0.0629	0.0229

Table 3.12: Statistical summary of variance-of-RCD-differences for daily crime distributions in 2013. More frequent crime types tend to have a lower variance mean and a smaller range.

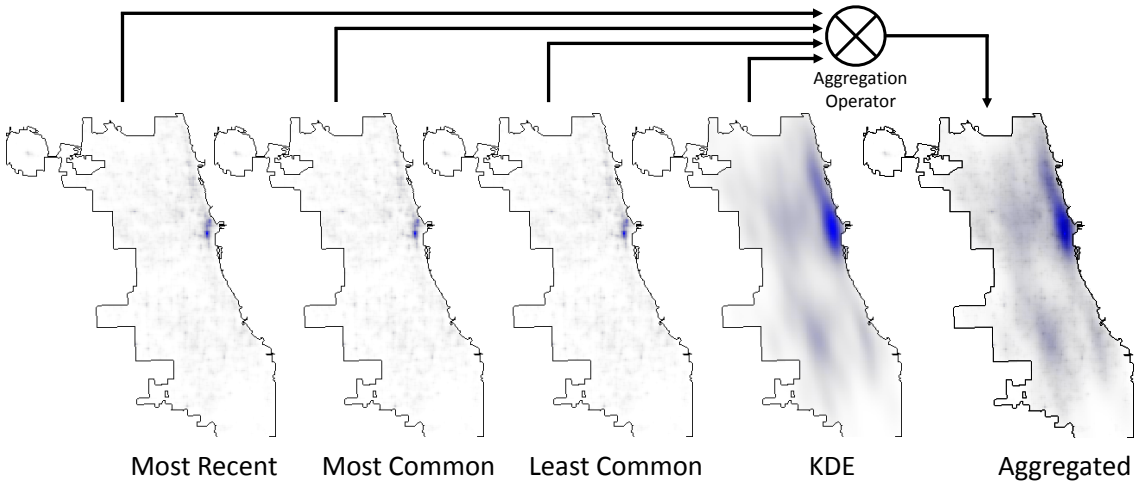


Figure 3.14: An illustration of my multi-kernel crime prediction approach where I use three training and evaluation windows to train three kernels. In this illustration, I show the map of Chicago, Illinois with four threat surfaces predicted by the three kernels and the KDE. I use an aggregation operator to fuse the surfaces and generate a final prediction surface.

3.5.3 Mathematical approach

From the variance analysis of daily crime distributions, I observe that the GA evaluating window (i.e., last three days prior to the testing day) used to train local kernels may not be representative of future incidents especially for infrequent crime types. In this section, I propose a new multi-kernel approach in which I choose better and more representative training and evaluation windows, train localized kernels, and aggregate the predicted threat surfaces using a fusion method based on crowd sourcing [70]. Figure 3.14 shows my approach for multi-localized kernel crime prediction process which involves: (1) I generate images using incidents from every three consecutive days within one year prior to the testing day. For example, if I am building a prediction model to predict thefts on 2013-09-29, I generate images using thefts from 2013-09-26 $\rightarrow$ 2013-09-28, 2013-09-25 $\rightarrow$ 2013-09-27, ..., 2012-09-27 $\rightarrow$ 2012-09-29; (2) I calculate RCD distances of the pair wise images and filter outliers; (3) I choose three evaluation intervals: most recent (R) (i.e., the last three days prior to the testing day), most common (M) (i.e., the three days with the least average RCD differences), and least common (L) (i.e., the three days with the largest RCD distance from the most common interval). Note, in all of the three cases, the training observations for the GA come

from one month data prior to the evaluation interval. For example, if the most common interval is 2013-05-10 to 2013-05-12, then I train the GA on incidents from 2013-04-10 to 2013-05-09. The motivation behind choosing the three intervals is that if the most recent interval will not be representative of future incidents, then the most common interval will have the highest chance of being representative. Also, if the distribution of future incidents varies from the most common distribution, then the least common interval would better represent those incidents. After I learn the three kernels, I generate localized kernel estimates using incidents from one month prior to the testing day. Finally, I compute the final predictions by aggregating the three localized kernel estimates using a data fusion method described in [70]. I call this model: multi-localized kernel density estimation (MLKDE). I also generate density estimates using the traditional KDE method and I average them with the estimates resulted from MLKDE. The motivation for doing so is that the three kernels included in the MLKDE are informed by local information while the density estimates produced by the traditional KDE are informed by global information. By taking the average of both methods, I can produce density estimates that are informed by both local and global information. This approach is especially advantageous in dealing with sparse crime types since very little information is available at lower resolutions. I call this model multi-localized and global kernel density estimation (MLGKDE).

3.5.4 Evaluation

I evaluated my multi-kernel approaches on real crime records from Chicago between 2013-08-28 and 2014-04-14. I followed the same experimental setup from Section 3.4.3: (1) I created a grid of 200-meter squares covering the official city boundary of Chicago; and (2) I enumerated 70 random testing days for each of the 17 crime type; (3) for each testing day and crime type, I performed a distribution analysis for incidents from one year prior to the testing day, and I obtained the most recent (R), most common (M) and least common (L) evaluation intervals; (4) I fitted one KDE and three LKDE models (namely, LKDE-R, LKDE-M and LKDE-L) using crime data prior to the testing day; (5) I used each of the fitted

models to predict hotspots for the testing day; (6) I aggregated the density estimates that are generated from LKDE-R, LKDE-M, and LKDE-L into a MLKDE model; (7) I averaged the final density estimates from MLKDE along with the ones from the traditional KDE; (8) Finally, I aggregated all testing days for a model and crime type following the process described by Gerber [10]. Figure 3.15 shows one of the seventeen aggregated surveillance plots. This plot illustrates the usefulness of my multi-kernel approach. Even though a lower performance is achieved using a single kernel, LKDE-R in this plot, the overall multi-kernel performance is increased with the help of the other kernels. Table 3.13 shows the AUC of the

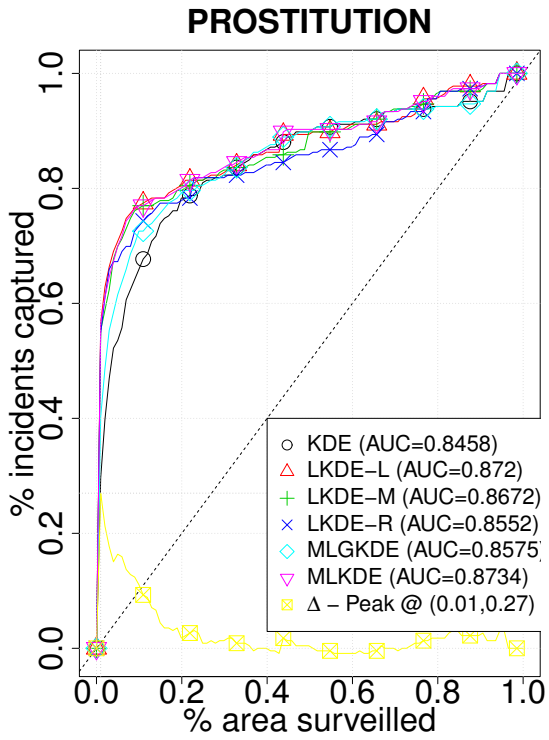


Figure 3.15: Aggregated surveillance plots of prostitution crimes. The gain series, shown in yellow, is computed as MLKDE-KDE. The highest peak gain occurs at the location specified by $\Delta - Peak$.

aggregated surveillance plots for six density estimation methods over 70 prediction days for 17 crime types, sorted by decreasing crime frequency. The performance of LKDE-R, LKDE-M, and LDKE-L confirm the findings of the variance analysis from Section 3.5.2 and support motivation for having a multi-kernel approach for density estimation. First, none of the

kernels outperformed the other two or the traditional KDE with respect to all crime types. Second, the variance in performance between LKDE-R, LKDE-M, and LDKE-L increases as the frequency of the crime types decreases. As for the aggregated methods, KDE was outperformed in all crime types by either one or both methods. MLKDE outperformed KDE for 15 out of 17 crime types, often achieving substantial gains. Three out of the fifteen types, namely interference with public officer, public peace violation, and weapon violation, are infrequent crime types. In these crime types, the previous LKDE method (i.e., LKDE-R) was outperformed by the traditional KDE. This shows the advantage of aggregating density estimates generated by multiple kernels. Also, adding the most common and the least common evaluation intervals improved the robustness of the MLKDE. When one of the two is under-performing in some crime types, the other one is achieving a better performance which ultimately helps in sustaining an overall good performance for the aggregated MLKDE model. For example, when predicting interference with public officer incidents, LKDE-L had a significantly lower performance than the other models including KDE. However, with the help from the LKDE-M, MLKDE achieved a higher performance and was able to outperform the KDE. Moreover, the MLGKDE model outperformed the KDE and achieved the highest AUC for 5 out of 6 most infrequent crime types. This shows how incorporating both global and local information helped in leveraging the sparsity challenge for localized estimation methods. Finally, the insights that multiple kernels provide can help criminologists in understanding variations in crime patterns between different crime types. For example, the fact that the kernel fitted on most recent incidents would outperform the ones fitted on most common and least common incidents could indicate that future incidents of this crime type would most likely follow its recent distribution. Similar insights can be drawn from most common and least common kernels where future spatial distribution of incidents can be either consistent or inconsistent with the historical distribution.

Crime Type	KDE	LKDE-R	LKDE-M	LKDE-L	MLKDE	MLGKDE
THEFT	0.7156	0.7860	0.7872	0.7880	**0.7880**	0.7576
BATTERY	0.7269	0.7923	0.7941	0.7946	**0.7946**	0.7803
NARCOTICS	0.8331	0.8749	0.8755	0.8759	**0.8763**	0.8659
CRIMINAL DAMAGE	0.6725	0.7275	**0.7300**	0.7253	0.7298	0.7212
OTHER OFFENSE	0.6937	0.7298	0.7389	**0.7390**	0.7382	0.7268
BURGLARY	0.6998	0.7403	0.7446	**0.7466**	0.7463	0.7363
ASSAULT	0.7204	0.7559	**0.7607**	0.7521	0.7598	0.7578
DECEPTIVE PRACTICE	0.7433	0.7657	0.7724	0.7735	**0.7736**	0.7700
MOTOR VEHICLE THEFT	0.6987	0.7125	0.7173	0.7184	0.7214	**0.7224**
ROBBERY	0.7463	0.7768	**0.7802**	0.7762	0.7801	0.7764
CRIMINAL TRESPASS	0.7592	0.7991	0.7969	0.7950	**0.8027**	0.7942
WEAPONS VIOLATION	0.7826	0.7764	0.7705	0.7835	0.7845	**0.7890**
PUBLIC PEACE VIOLATION	0.7401	0.7342	0.7337	0.7570	0.7517	**0.7615**
OFFENSE INVOLVING CHILDREN	0.6837	0.6722	0.6664	0.6806	0.6826	**0.6894**
SEX OFFENSE	0.6745	0.6422	0.6660	0.6458	0.6600	**0.6793**
PROSTITUTION	0.8458	0.8552	0.8672	0.8720	**0.8734**	0.8575
INTERFERENCE WITH PUBLIC OFFICER	0.7889	0.7810	0.7934	0.7470	0.7947	**0.7994**

Table 3.13: Performance of six density estimation methods on 70 random prediction days with respect to the 17 crime types.

3.5.5 Conclusions

I have extended a previously proposed localized kernel density estimation, and proposed two multi-kernel approaches. I combine density estimates generated by three different kernels fitted on most recent, most common and least common days. I further average the fused density estimates with the ones generated by a traditional KDE to produce density estimates that are informed by both local and global information. My experiments on real crime data of 17 types from Chicago shows the effectiveness of my approach versus traditional KDE and single LKDE. Multi-kernel density estimations approaches outperformed the KDE and single LKDE in all crime types, and including the global information in the fusion process is shown to be advantageous especially for sparse crime types. Most importantly, the experiments show the robustness of my multi-kernel methods with respect to different crime types that vary in nature and/or frequency, and for which a single kernel is insufficient for making high quality forecasts. Furthermore, the difference in performance between the three single kernels may provide insights into the future spatial distribution of incidents with respect to specific crime types. These insights would help criminologists better understand or explain spatial crime patterns and how they change over time.

Crime Prediction Models Evaluation: The Good, the Bad and the Ugly

Researchers have proposed a number of evaluation metrics [10, 18, 63, 71] for crime prediction. In this section, I present three evaluation methods: predictive accuracy index [18], predictive efficiency index* [71], and surveillance plots [10]. I compare and contrast the three methods. I also explore the pitfalls of some of these methods and how they can be greatly affected by the settings of the prediction problem. To the best of my knowledge, there has not been a comprehensive study about crime prediction evaluation metrics presented in the literature before. Therefore, my work here bridges the academic gap and provides guidance for practitioners about the situations under which the use of such evaluation methods would be mostly appropriate.

4.1 Predictive accuracy index

Chainey et al. proposed the predictive accuracy index (PAI) as a way of measuring the effectiveness of predictions. PAI is calculated as the hit rate (the ratio of incidents occurring within hotspots to the total number of incidents) divided by the percentage of all area covered by the hotspots [18]. Formally, PAI is defined as

$$\text{PAI} = \frac{\frac{n}{N}}{\frac{a}{A}} \tag{4.1}$$

where a is the predicted area to be of high threat (i.e., a hotspot), and A is the area of the entire study region, n is the number of incidents that occur in the predicted area, N is the total number of crimes that occur in A. For example, if the selected hotspots covered 20% of the study region and 30% of future crime fell into these hotspots, the PAI would be $\frac{30\%}{20\%} = 1.5$. PAI is thus higher when more crime is present within a smaller set of predicted hotspots.

Although intuitive, PAI relies on the selection of an area-coverage parameter for the hotspots. This selection can be difficult to justify, and it is not straightforward to visualize and relate PAI scores for multiple area-coverage selections. More practically, if it turns out that the police have resources sufficient to patrol an area that is smaller or larger than the selected area-coverage size, then the PAI that was calculated will not be an accurate indication of performance when guided by the hotspot map.

4.2 Predictive efficiency index*

Hunt proposed the predictive efficiency index* (PEI*) as an approach for measuring the efficiency of the predictions [71]. PEI* is a complementary measure to the PAI and it is meant to measure how well a prediction does compared to how well it could have done. PEI* is defined as

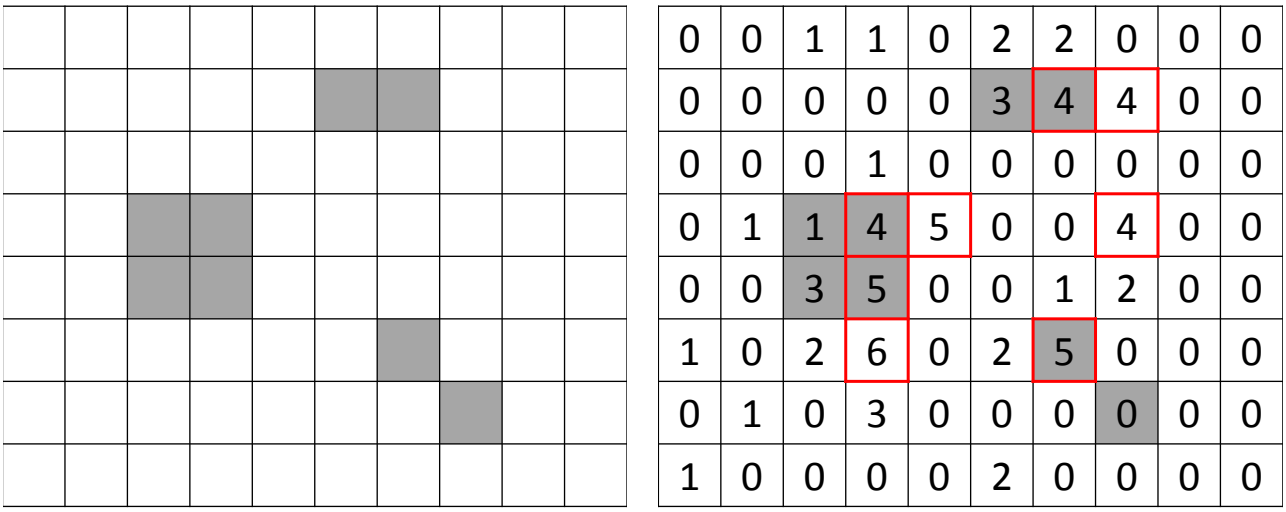
$$\text{PEI*} = \frac{\text{PAI}}{\text{PAI*}} \tag{4.2}$$

where PAI* is the maximum obtainable PAI value for the same predicted area a. In another words, by fixing a and knowing that N and A are fixed, then PEI* equals

$$PEI* = \frac{PAI}{PAI*} = \frac{\frac{\frac{n}{N}}{\frac{a}{A}}}{\frac{\frac{n*}{N}}{\frac{a}{A}}} = \frac{n}{n*} \tag{4.3}$$

where $n*$ is the largest number of future incidents that will occur in any region whose area equals to a.

Example: Consider that a crime prediction model used to forecast future crime in a certain area of interest, shown in Figure 4.1a. The model has forecasted 8 grid cells, as shown shaded in gray color in Figure 4.1a, to have the highest threat (i.e., hotspots). After future crime incidents during the testing window had occurred and recorded (as shown in Figure 4.1b, I can evaluate the performance of the prediction model using PAI and PEI*. First, out of 80 cells, only 8 cells were predicted as hotspots. Therefore, $A = 80$ and $a = 8$. Also, the total number of testing incidents is $N = 67$. Next, I count the number of incidents that occurred in the



0	0	1	1	0	2	2	0	0	0
0	0	0	0	0	3	4	4	0	0
0	0	0	1	0	0	0	0	0	0
0	1	1	4	5	0	0	4	0	0
0	0	3	5	0	0	1	2	0	0
1	0	2	6	0	2	5	0	0	0
0	1	0	3	0	0	0	0	0	0
1	0	0	0	0	2	0	0	0	0

(a) Predicted hotspots (b) Counts of testing incidents

Figure 4.1: Example of an 8 predicted hotspot cells (shaded in gray color) along with future incidents that occur in the testing window. Cells highlighted in red color are the 8 cells with the largest number of incidents. This example illustrates the computation of PAI, PAI*, and PEI* metrics.

predicted hotspot cells which equals to $n = 25$. Then, PAI $= \frac{\frac{25}{67}}{\frac{8}{80}} = 3.73$. Next, to measure the PEI*, I need to count the largest number of testing incidents obtainable by any 8 cells. Figure 4.1b shows that the 8 cells with the largest testing incidents (highlighted in red) include $n* = 37$ testing incidents. Therefore, PAI* $= \frac{\frac{37}{67}}{\frac{8}{80}} = 5.52$ and PEI* $= \frac{3.73}{5.52} = \frac{25}{37} = 0.676$.

Like the PAI, PEI* relies on the selection of an area-coverage parameter, a, for the hotspots. Therefore, it has the practical disadvantage of not knowing the actual prediction performance in case the police has resources sufficient to patrol an area that is smaller or larger than the selected area-coverage size.

Later in this chapter, I address the limitation of PAI and PEI* by introducing three plots that generalize the metrics to include performance at all area-coverage values, but first, I will discuss the relation between PAI and PEI*, and how their values can be affected by changing the grid cell size. I will also perform a performance analysis to show the trade-off between both metrics.

4.3 PAI vs PEI*: performance analysis

Researchers have been rigorously studying the hotspot analysis and its parameterization [15, 18, 21, 22, 24, 63–66]. With regard to grid cell size in particular, researchers such as Chainey have come to the conclusion that it has a small effect on the density estimation [65]. Although this conclusion might be true for density estimation, it is not the case for PAI and PEI*. I will show in this section that varying the cell size has a significant impact on both evaluation metrics.

0	0	1	1	0	2	2	0	0	0
0	0	0	0	0	3	4	4	0	0
0	0	0	1	0	0	0	0	0	0
0	1	1	4	5	0	0	4	0	0
0	0	3	5	0	0	1	2	0	0
1	0	2	6	0	2	5	0	0	0
0	1	0	3	0	0	0	0	0	0
1	0	0	0	0	2	0	0	0	0

(a) Grouping four cells into one

0	2	5	10	0
1	6	5	4	0
1	16	2	8	0
2	3	2	0	0

(b) Counts of grouped testing incidents

Figure 4.2: A continuation of example in Figure 4.1 such that the grid cell size in increased by 4 times. With the area of the predicted hotspot regions, a, remaining the same, changing the cell size has a great impact on the values of PAI and PEI*.

Before I study the effect of changing the cell size on performance metrics, I revisit the example in Figure 4.1 to illustrate the concept of grid cell size, and how changing it would alter the underlying distribution of criminal incidents. In this example, I am interested in increasing the cell size by 4 times, i.e., every four cells from the previous settings will be grouped into a single cell in the new one (as shown in Figure 4.2a). This change will reduce the resolution of the study region. For example, the grouped testing counts at cells (1,3) and (2,3) in Figure 4.2b indicate that both have the same importance since an equal number of

incidents (5) have occurred at both cells. However, after we look at the incidents counts using the smaller cell size in Figure 4.2a, it is clear that both cells are not of equal importance. All of the five incidents in cell (2,3) have occurred only at one-fourth of the area while the ones in (1,3) were split between two smaller sub-cells. To conclude, the bigger the cell size is, the less information about the actual crime distribution I will have because incidents counts are being aggregated at a lower resolution. In theory, the ideal configuration for this problem is to use a continuous prediction model where forecasts are being made at a pinpoint resolution. However, the continuous representation has two major limitations. First, by making forecasts at pinpoint resolution and predicting a set of GPS locations to be of high risk, it become more challenging to define risk. How about immediate points surrounding the predicted locations? Should they be considered as high risk points as well? How about points that are 10 meters away for the predicted locations? How about points at a 100 meter distance? Using a continuous representation, it is hard to define a specific distance threshold after which, points would have low risk. The only way around this challenge is to make a prediction for each and every point in the study region, and the problem become computationally intractable. The second limitation with the continuous representation is that police officers while patrolling areas will have a deterrent effect over areas rather than pinpoint locations. Therefore, using continuous forecasts, it is hard to allocate police resources and analyze their impact on crime prevention.

To overcome the computational and practical limitations of continuous representation, practitioners discretize the study region and build prediction models accordingly. Next, I will compute the PAI and the PEI* for the prediction with the larger cell size in Figure 4.2b. The total area of the forecasted hotspots equals to the one in the previous example in Figure 4.1, i.e., $a = 2$ and $A = 20$. Also, the total number of testing incidents is fixed (i.e., $N = 67$). However, the number of testing incidents occurring within the predicted hotspots as well as the maximum obtainable number of incidents have changed. The former is $n = 22$ while the latter is $n* = 26$. Therefore, PAI $= \frac{\frac{22}{67}}{\frac{2}{20}} = 3.28$; PAI* $= \frac{\frac{26}{67}}{\frac{2}{20}} = 3.88$; and

PEI* $= \frac{3.28}{3.88} = \frac{22}{26} = 0.846$. In this example, I observe that increasing the grid cell size by a factor of 4 to 1 leads to an increase in the PEI* at the expense of the PAI. This is also the case even if the predicted area remain unchanged regardless of cell size, as shown in Figure 4.3, a larger cell size will increase the PEI* (PEI* = 0.76 for small cell size; PEI* = 1 for large cell size) and while the PAI remains unaffected (i.e., PAI = 4.77 for both configurations). The reason for this difference is that the maximum obtainable number of future incidents, $n*$, will increase as the cell size decreases and vice versa. For example, in Figure 4.2b, cell (2,3) has 5 incidents, and in case it would be included as part of the $n*$ summation, then its entire area will account only for 5 incidents. However, these incidents come only from one-fourth of cell (2,3) (see Figure 4.2b). As a result, if the smaller cell is used, three-fourth of that cell area will be reallocated to other areas with higher incidents count leading to an increase in the summation of $n*$.

0	0	1	1	0	2	2	0	0	0
0	0	0	0	0	3	4	4	0	0
0	0	0	1	0	0	0	0	0	0
0	1	1	4	5	0	0	4	0	0
0	0	3	5	0	0	1	2	0	0
1	0	2	6	0	2	5	0	0	0
0	1	0	3	0	0	0	0	0	0
1	0	0	0	0	2	0	0	0	0

(a) Small cell size

0	2	5	10	0
1	6	5	4	0
1	16	2	8	0
2	3	2	0	0

(b) Large cell size

Figure 4.3: An example of a prediction problem where the predicted area, a, remains fixed in lieu of grid cell size. Nonetheless, varying the cell size would still affect the PEI* due to the difference in $n*$.

Making a decision on which evaluation metric to optimize will control the choice of the grid cell size. If designers want to increase the PAI, they need to choose a smaller cell size. In this case, algorithms will have more flexibility in predicting hotspot areas that would

capture the future crime distribution. On the other hand, if practitioners are interested in increasing the PEI*, then a larger cell size need to be chosen. Although this choice might lead to a smaller number of captured incidents, n, it will obtain a higher PEI* because $n*$ will also be reduced. Nonetheless, in more realistic scenarios, practitioners would be interested in optimizing both metrics. In such cases, experimental performance analyses need to be conducted in order to find the cell size that would provide the best trade-off between PAI and PEI*. A number of factors can impact these analyses such as the crime type and its frequency, the prediction algorithm, the area of interest, etc. In this section, I will use the city of Portland, Oregon, United States as a test-bed. I preform a performance analysis on all call-for-service (CFS) between March 2013 and December 2016. I obtained geotagged CFS data from National Institute of Justice (NIJ), Real-Time Crime Forecasting Challenge.[1] The challenge included incidents of more than 18 different types between 2012-03-01 and 2017-02-28. More detailed about the data and NIJ challenge are described in Chapter 6. As for the PAI vs PEI* performance analysis, I used the following experimental settings: 1) I set the testing window to three months and the training window to one year prior to the testing window (e.g., predicting all CFS between 2013-03-01 and 2013-05-31 using observations from 2012-03-01 to 2013-02-28); 2) I modeled the problem using LKDE models from Section 3.4; 3) I ran 15 predictions for consecutive three-month testing window starting from 2013-03-01→2013-05-31 and ending with 2016-09-01→2016-11-30; 4) I varied the grid cell size between 350 and 600 feet with increments of 50; and 5) I fixed the total sum of predicted areas as hotspots to be equal to 0.25 square miles (i.e., with different cell sizes, different number of cells will be forecasted as hotspot. Nevertheless, the summation of all of these cells' area is about 0.25 square miles regardless of the cell size). Table 4.1 shows the number of cells predicted as hotspots with respect to different cell sizes. Figure 4.4 shows the average number of incidents that occurred in forecasted hotspot cells (n), and the average maximum-obtainable number of incidents from any forecasted cells ($n*$) as I vary the cell

[1]NIJ Real-Time Crime Forecasting Challenge: https://nij.gov/funding/Pages/fy16-crime-forecasting-challenge.aspx

Cell Size	Number of hotspot cells	Total forecasted area
350	57	6,982,500 sq ft (0.250 sq mi)
400	44	7,040,000 sq ft (0.252 sq mi)
450	35	7,087,500 sq ft (0.254 sq mi)
500	28	7,000,000 sq ft (0.251 sq mi)
550	24	7,260,000 sq ft (0.260 sq mi)
600	20	7,200,000 sq ft (0.258 sq mi)

Table 4.1: Number of hotspot cells forecasted with respect to different cell size configurations.

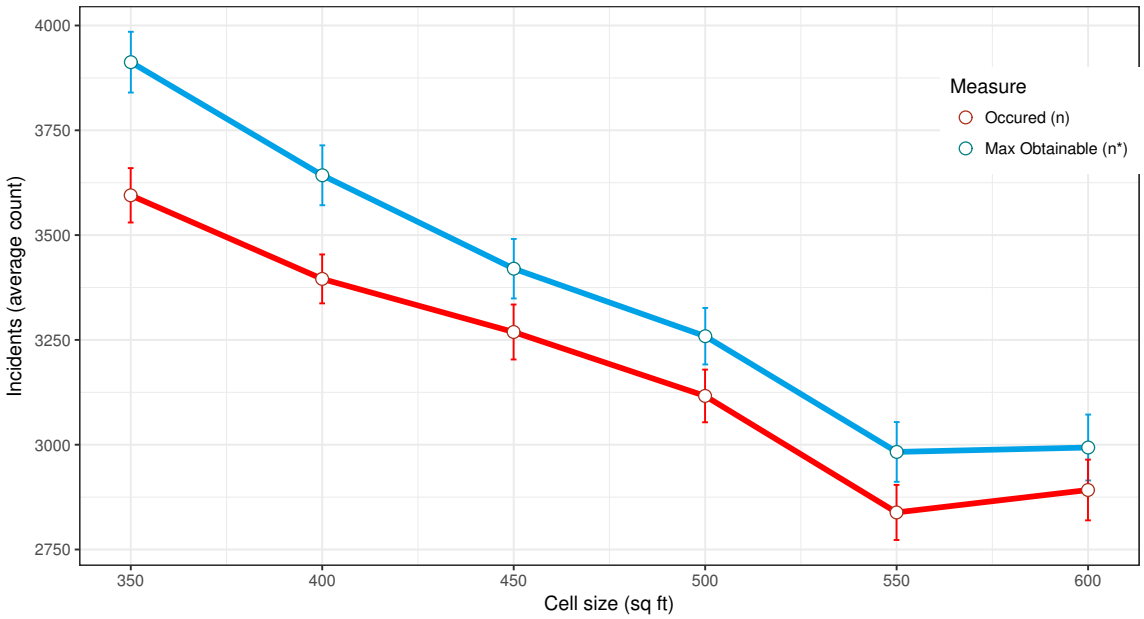


Figure 4.4: Average number of testing incidents (n) captured by 15 three-month LKDE prediction models on all call-for-service from Portland between 2013-03-01 and 2016-11-30, and average maximum-obtainable incidents ($n*$) given the forecasted area by the prediction models. With smaller cell sizes, both n and $n*$ increase.

size. The grid cell size has an inverse relation to both n and $n*$ such that they both decrease as the cell size increases. This confirm the previous observation from Figures 4.2 and 4.3 that with a smaller cell size, I have more flexibility in covering more hotspot areas, and thus, increasing the captured incidents count. Note, the gap between n and $n*$ increases as smaller cell sizes are used. The reason for this increase is that the prediction problem become harder when a smaller cell size is used since more cells need to be forecasted as hotspots. On the other hand, Figure 4.5 shows the PAI, PAI* and PEI* performance trends with respect to different cell sizes. PAI and PAI* follow a similar pattern to n and $n*$ such that they increase

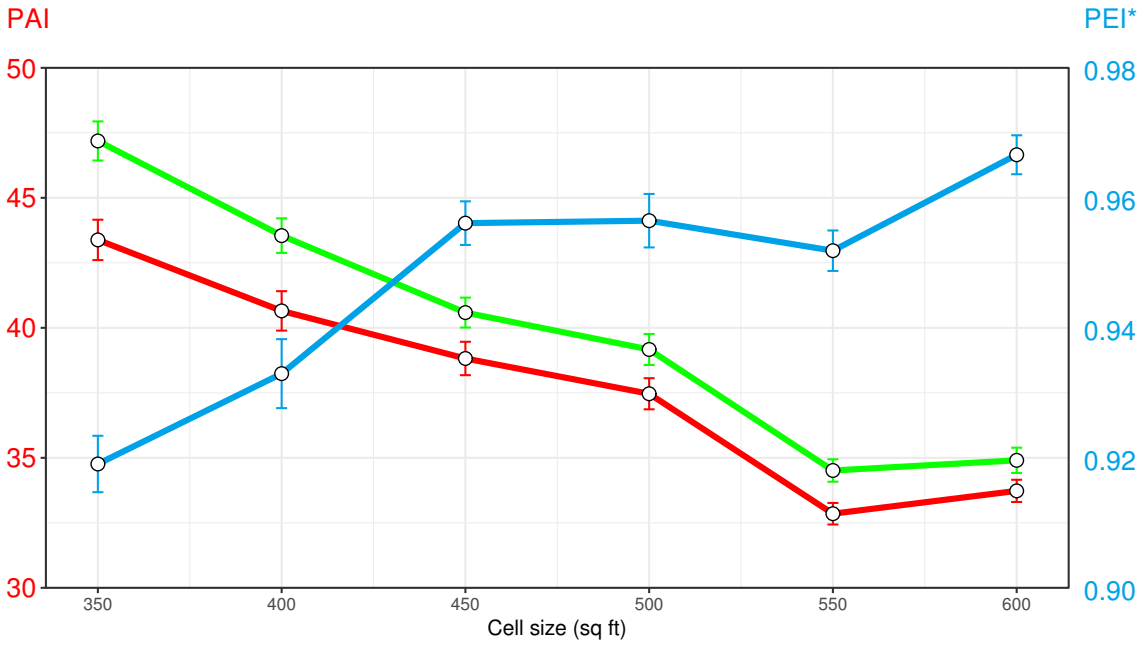


Figure 4.5: Comparison of PAI (shown in red), PAI* (shown in green) and PEI* (shown in blue) with different grid cell sizes. PEI* has an inverse relation with PAI and PAI*, and using a smaller cell size will help in achieving a higher PAI. Note, all values are averaged over 15 three-month LKDE predictions on all call-for-service from Portland between 2013-03-01 and 2016-11-30.

as the cell size decreases, and the gap between PAI and PAI* become smaller when a larger cell size is used. As for the PEI*, with a smaller cell size, a lower PEI* is obtained because of the increase in the gap between PAI and PAI*.

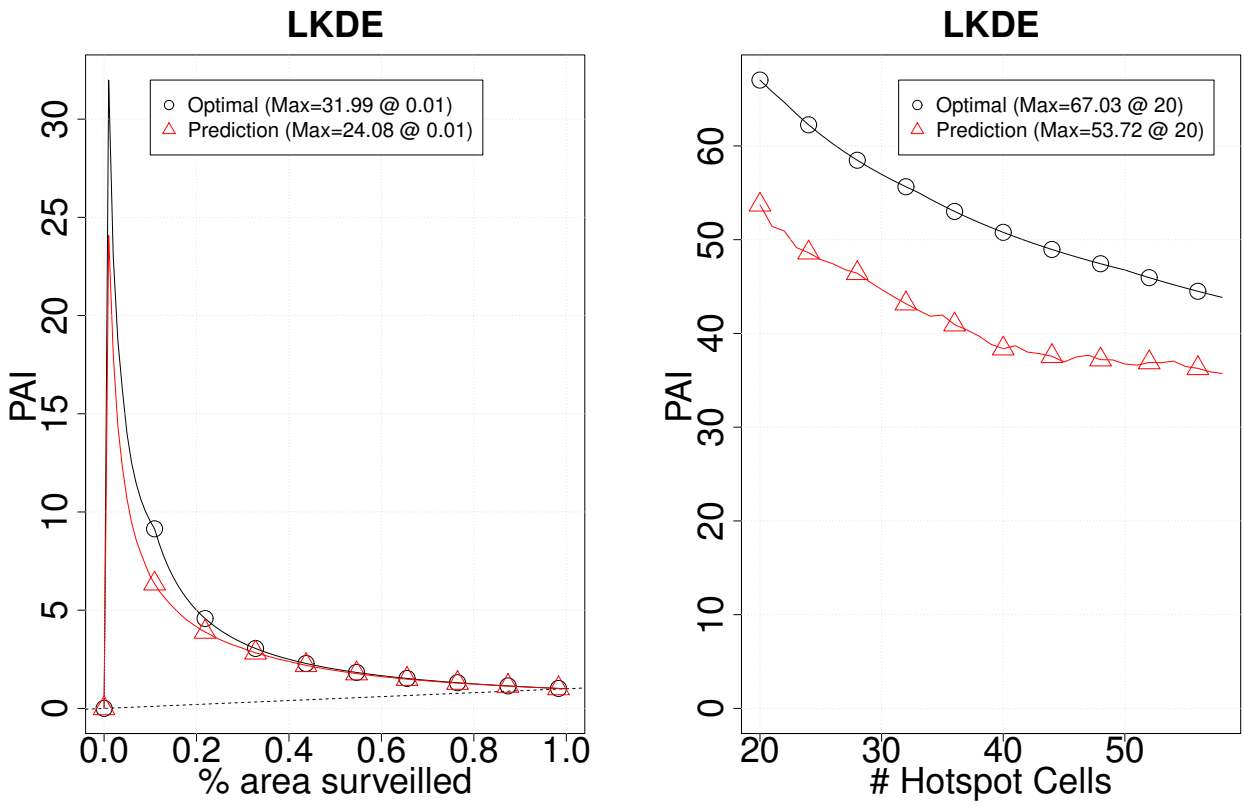
Practitioners can conduct a performance analysis to find the appropriate cell size that would provide the best trade-off between PAI and PEI*. For example, by reducing the cell size from 500 to 450, we would observe an increase of 152 in n, and 1.36 in PAI with almost no decrease in the value of PEI*.

4.4 PAI and PEI* plots

The main challenge for using the PAI and PEI* is that they depend on the area-coverage parameter for the hotspots. As police resources vary over time and space, it become harder to evaluate prediction models using metrics that are fixed in both space and time. To address the spatial limitation, I introduce two PAI and PEI* plots which generalize the PAI and the

PEI* to include performance at all area-coverage values. The new plots help practitioners to evaluate and compare models on areas that are smaller or larger than the selected hotspot area. The only requirement for creating these plots is to relax the binary assumption of hotspots (i.e., a grid cell has a value of 1 if it is a hotspot, and 0 otherwise). Instead, prediction models need to calculate a numerical value for each cell such as a likelihood of crime occurrence, a threat score, a risk score, etc. Then, grid cells are sorted and surveilled accordingly. Figures 4.6 and 4.7 show examples of PAI and PEI* plots. These plots show the PAI, PAI* and PEI* values (y-axis) at each x% most threatened areas. For example, Figure 4.6a shows that if I consider the most threatened 20% area as predicted by a LKDE model, then this model would achieve a PAI value of 4.5. Also, the PAI plot shows that the maximum PAI value is obtained at 1% of the area, and knowing that PAI is a measure of accuracy, then if the police patrol more than 1% according to the forecasts generated by this model, they will not improve upon the accuracy. In another words, if we consider police resources as an investment for a return on improving public safety, then the best return on investment is achieved at 1% area. Likewise, Figure 4.7a shows that, for example, following the model forecasts to patrol the most threatened 20% area, a PEI* value of 0.82 would be achieved. In addition, PEI* highlights the point at which $n* = N$ such that all future incidents are captured within the corresponding most threatened x% as predicted by the model. This point indicates the area-percentage that is required to cover all future incidents, and assuming we have an optimal model (i.e., $n = n*$), then we would achieve a hit-rate of 1 at that point.

The main limitation for the PAI and PEI* plots is that their values are computed with respect to a percentage of area. Therefore, only 100 PAI, PAI* and PEI* values are calculated, and the values at other area-percentages are not included in the plots (e.g., the plots would include the PAI, PAI* and PEI* at 1% but they will be missing the values for 0.5%, 0.1%, etc.). To overcome this limitation, I provided another version of the plots (shown in Figure 4.6b and 4.7b) that is focused on the most threatened 20 to 60 grid cells. For example, the



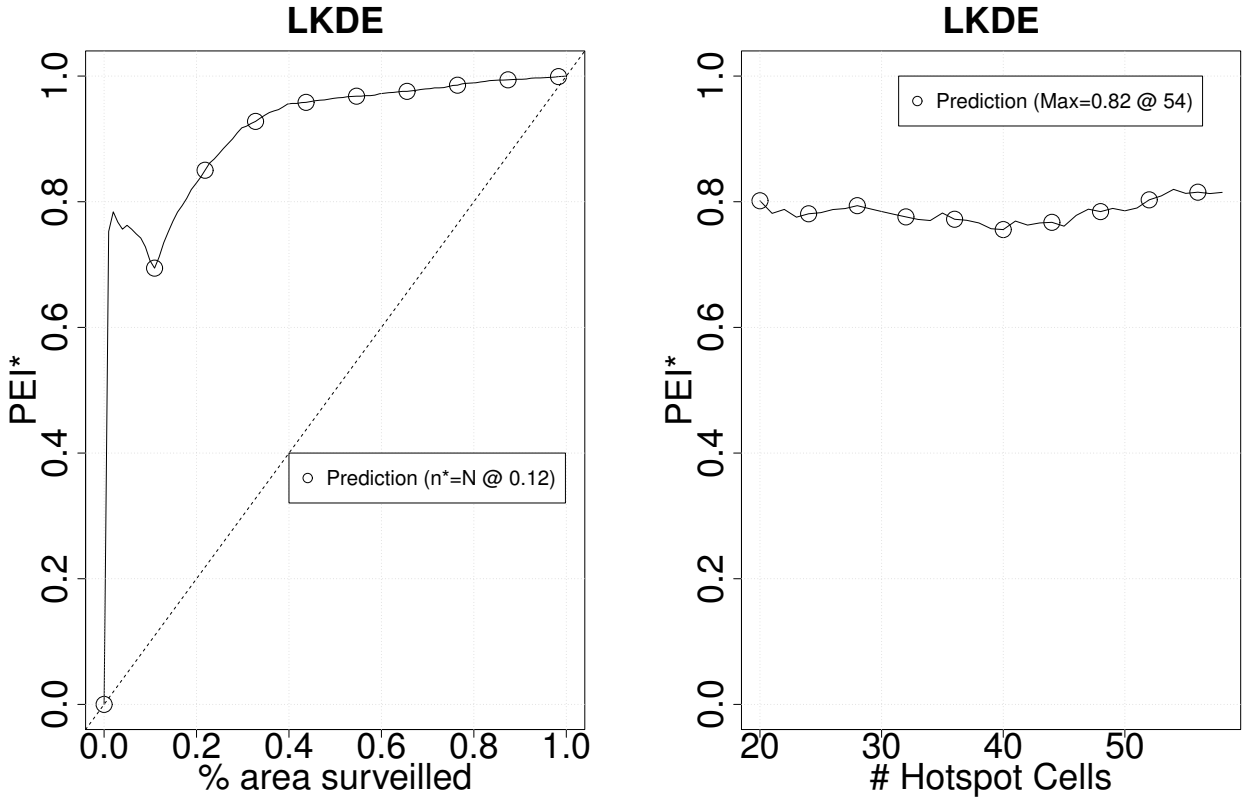
(a) (b)

Figure 4.6: (a) A PAI plot, which shows the PAI values (y-axis) that correspond to the most threatened x% area predicted by the model (x-axis). The optimal series shows the maximum obtainable PAI (i.e., PAI*) at x% area. (b) A zoomed version of the plot in (a) that shows the PAI and PAI* values for the most threatened 20 to 60 grid cells.

zoomed version in Figure 4.6b shows that the LKDE model achieves a PAI of 53.72 if the most threatened 20 cells are considered as hotspots. These cells constitute about 0.17% of the area of Portland, and the PAI value at that percentage will not be included in the full plot in Figure 4.6a. A similar observation can be made from the plot in Figure 4.7b. Therefore, zoomed PAI and PEI* plots are used complementary to the full plots in case practitioners are interested in looking at the performance within a small range of areas.

4.5 Surveillance plots

Another evaluation plot, proposed by Gerber [10], is surveillance plots. A surveillance plot shows the proportion of true future crimes, i.e., $\frac{n}{N}$, (y-axis) that occur within the x%



(a) (b)

Figure 4.7: (a) A PEI* plot, which shows the PEI* values (y-axis) that correspond to the most threatened x% area predicted by the model (x-axis). The x% area on which the maximum obtainable number of incidents ($n*$) equals to the total number of incidents (N) is also highlighted. (b) A zoomed version of the plot in (a) that shows the PEI* values for the most threatened 20 to 60 grid cells.

most threatened area predicted by the model (x-axis). Figure 4.8 shows that around 70% of future crime incidents (y-axis) are captured by the top 11% most threatened area (x-axis), as predicted by a LKDE model. The curves for optimal predictions approach the top-left corner, indicating that the predicted crime risks reflect the true spatial distribution of crime. I summarize surveillance plots with scalar area under the curve (AUC) values. Surveillance plots are similar, in concept, to traditional ROC curves, but offer two advantages over the ROC curves for the crime prediction task. First, surveillance plots provide decision makers with a better visualization of the underlying physical environment and crime process, as the x-axis is tied directly to the physical space being modeled. Second, traditional ROC curves

assume that the set of positive and negative instances is known with certainty. In the crime domain, the set of positive points is well defined: it is the set of points at which a crime occurred and was recorded by the police. The set of negative crime points is problematic. If a crime was reported at a specific latitude-longitude location, should a point 5 meters away from this location be given a ground-truth negative (NULL) label? What about 10 meters or 100 meters? The absence of a justifiable distance threshold for negative points renders the negative ground-truth indeterminate and precludes the application of ROC plots. Surveillance plots capture the same intuition as ROC plots (i.e., assign positive points higher probabilities than negative points) but rely only on the set of ground-truth positive points.

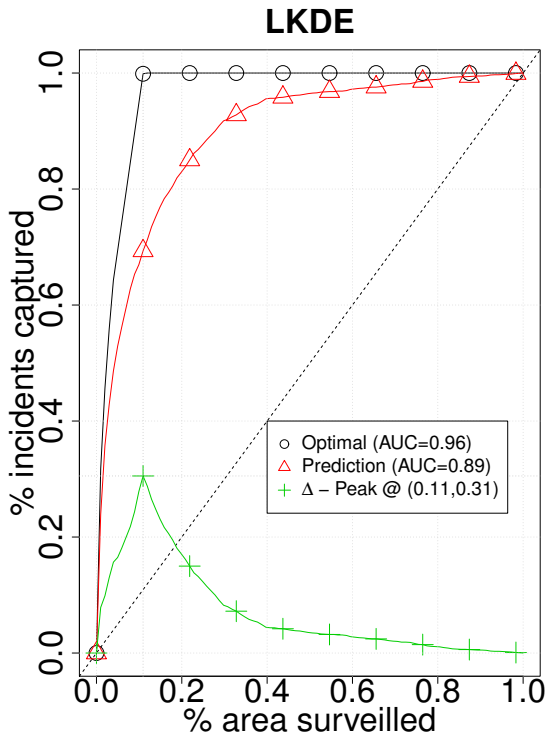


Figure 4.8: A surveillance plot, which shows the proportion of true future incidents (y-axis) that occur within the most threatened area predicted by the model (x-axis). This surveillance plot is summarized by its area under the curve (AUC) value.

Like the PAI and PEI* plots, surveillance plots generalize the PAI and PEI* metrics to include performance at all area-coverage values, which are depicted in a series. The x-values in this series are the area-coverage percentages, and the y-values in this series are the hit rates

associated with the x% "hottest" area as predicted by the model. For a given x-y coordinate in a surveillance plot, the PAI is thus $\frac{y}{x}$. Also, surveillance plots are analogous to PEI* plots in the way y values are computed. In PEI* plots, y is the PEI* value at the corresponding x percentage, i.e., $\frac{n}{n*}$ associated with the x% "hottest" area as predicted by the model. On the other hand, in surveillance plots it is computed as $\frac{n}{N}$. Therefore, starting from the x% at which $n* = N$, PEI* and surveillance plots become identical. In addition, surveillance plots support a natural resource allocation process: If public safety officials determine that they have patrol resources sufficient to cover x% of their area, the surveillance plot presents the anticipated crime hit-rate (y) at this coverage as well as potential hit-rate gains and losses resulting from more and less investment in the patrol activity (moving right and left along the surveillance series). I calculate the area under the curve (AUC) as a scalar summary of performance in order to rank multiple surveillance plots. Formally, the AUC is defined as

$$AUC = \sum_{i=1}^{P} \left((\widetilde{x}_i - \widetilde{x}_{i-1}) * \frac{\widetilde{y}_i + \widetilde{y}_{i-1}}{2} \right), \tag{4.4}$$

where $\widetilde{x}$ is the cumulative percentage of most threatened area predicted by the model, $\widetilde{y}$ is the accumulative percentage of testing incidents captured within the surveilled area such that $\widetilde{x}_0 = \widetilde{y}_0 = 0$, and P is the total number of performance points included in the surveillance plot. For example, each surveillance plot in Figure 4.8 include $P = 100$.

Finally, the evaluation plots, presented in this section, can be used to compare the performance of different models at different area percentages. Figure 4.9 shows an example comparison between LKDE and KDE models on street crimes in Portland between 2017-01-01 and 2017-01-31. In this comparison, and by observing the differences in performance, I conclude that the LKDE model has better accuracy (PAI), efficiency (PEI*), and surveillance performance (AUC). Also, the zoomed version of the plots, shown in Figure 4.10 shows that the peak accuracy for both models is achieved at the hottest 20 hotspot cells. Likewise, the efficiency of the KDE model drops as more cells are considered as hotspots. On the other hand, LKDE preserve its efficiency throughout the 60 most threatened grid cells.

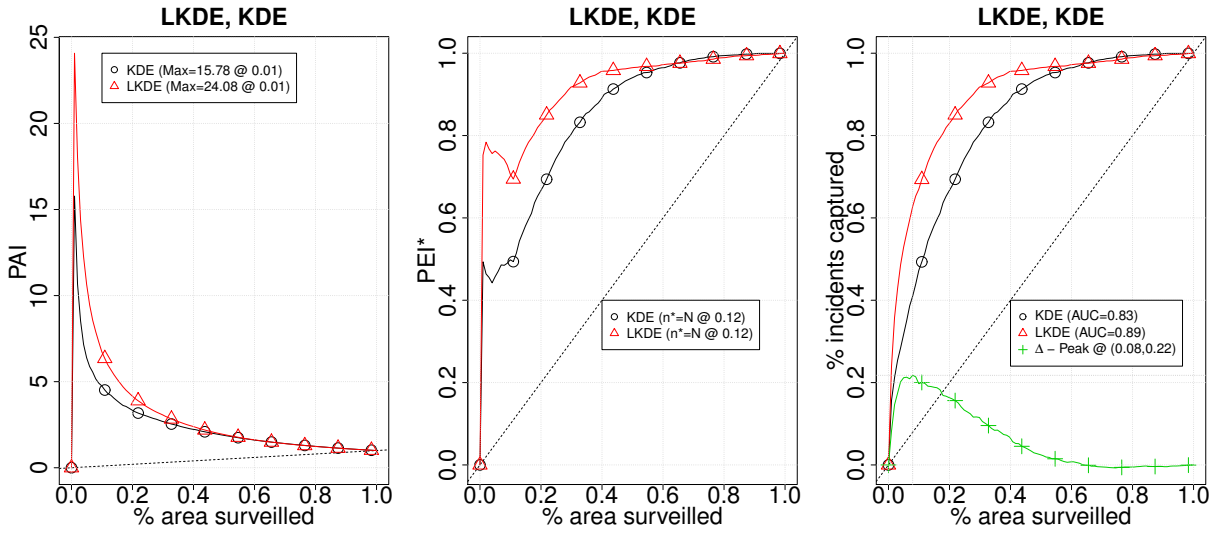


Figure 4.9: Comparison between LKDE and KDE models on street crimes in Portland for January 2017 using PAI, PEI*, and surveillance plots.

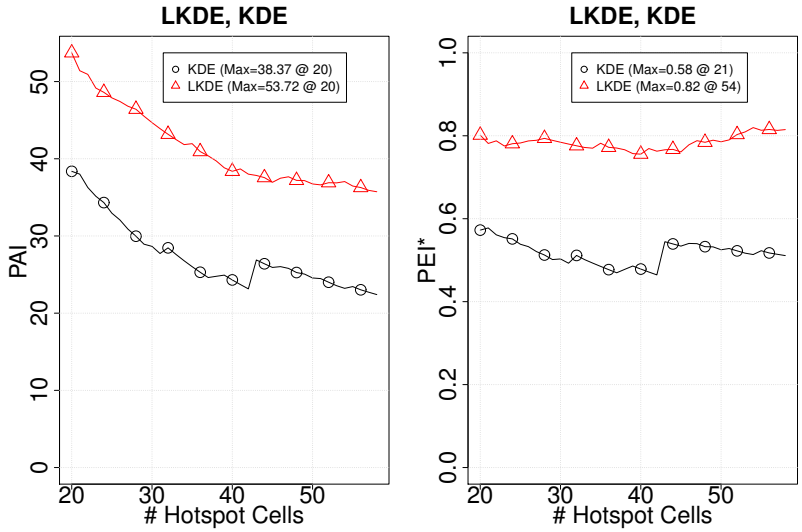


Figure 4.10: A zoomed PAI and PEI* plots for the most threatened 20 to 60 grid cells comparing the performance of LKDE and KDE models on street crimes in Portland for January 2017.

4.6 Conclusions

Practitioners and researchers have used two evaluate metrics, namely PAI and PEI*, to evaluate the accuracy and efficiency of crime prediction models. I presented a performance analysis between both metrics and show that varying the grid cell size has a significant

impact on their values. With a smaller grid cell size, the model obtains a higher PAI and lower PEI* values. The reasons for the difference in performance include: 1) the change in the underlying crime distribution as incident counts are being aggregated into bigger cells, and 2) because with smaller cells, models have more flexibility in covering smaller hotspot areas. Therefore, in order to perform a fair comparison between crime prediction models, I recommend that the compared models need to use the same grid cell size. I Also presented an experimental case study to show how a trade-off between PAI and PEI* can be obtained using the appropriate cell size. Furthermore, I discussed the spatial limitation of PAI and PEI* metrics because of their reliance on the selection of an area-coverage parameter for the hotspots. I addressed this limitation by introducing two PAI and PEI* plots and comparing them to existing surveillance plots. The new plots include the performance of models at different area-coverage percentages which helps in making resource allocation decisions in order to maximize the accuracy and efficiency of prediction models.

Visualization: A Human Factor Study on Dynamic Hotspot Maps

5.1 Introduction

Hotspot maps are geo-spatial visualizations that show the density of specific elements such as crime incidents, demographics, weather events, etc. on a map. Practitioners such as police crime analysts, urban city planners, evacuation and rescue squads, etc. use Hot-spot maps to make decisions in space and time. However, in most cases, it is hard to perceive the temporal element from hotspots since they show the density at a fixed point of time. Several solutions to overcome this challenge include showing: (1) multiple hotspots snapshot at different timestamps; (2) showing animated time-lapse hotspots. There are a number of issues with both solutions. First, multiple hotspots would show the map a different fixed time intervals. However, this information within two subsequent hotspots would be missing in the approach. Also, in order to increase the time granularity, i.e., use shorter time intervals, I need to include more hotspots, and therefore, it would be harder to visualize them all on one screen and make decisions. On the other hand, showing a single animated hotspots map would solve the granularity issue. However, it requires heavy memory load since decision makers have to remember the shape of the heat maps at different time steps in order to make a good decision. The memory load issue is solved by showing multiple maps on the same screen in which decisions can be made by comparisons without any memory efforts. Moreover, the intuition shows that either solution would provide better decision making than static maps. However, there has been no experimental study to prove that. In this chapter, I propose a dynamic visualization tool that provides the trade-off between memory load and granularity. The new tool would allow the practitioners to view the map at different times and take snapshots for direct comparisons. Also, in this chapter, I conduct a human factor

experiment to evaluate the different visualization approaches: (1) Static Maps, (2) Multiple Maps, and (3) Dynamic Maps.

5.2 Dynamic hotspot maps

The main limitation of traditional hotspot maps is that they present information at a fixed point of time. Current solutions require much more memory load from users. In this section, I present a new interface for visualizing dynamic hotspot maps. The new interface provides a way for traversing the map overtime and observing both spatial and temporal hotspot patterns. Figure 5.1 shows the new dynamic visualization interface which include the following functions: (1) a slider to move between different days or time intervals, (2) a button to take snapshots from the map at various time intervals, (3) a pane to compare the taken snapshots which can be expanded by clicking on the down arrow in (5), and (4) a date filter which can be used to navigate to a specific date and time interval. The expanded pane, shown in Figure 5.2, help decision makers to make side by side comparisons.

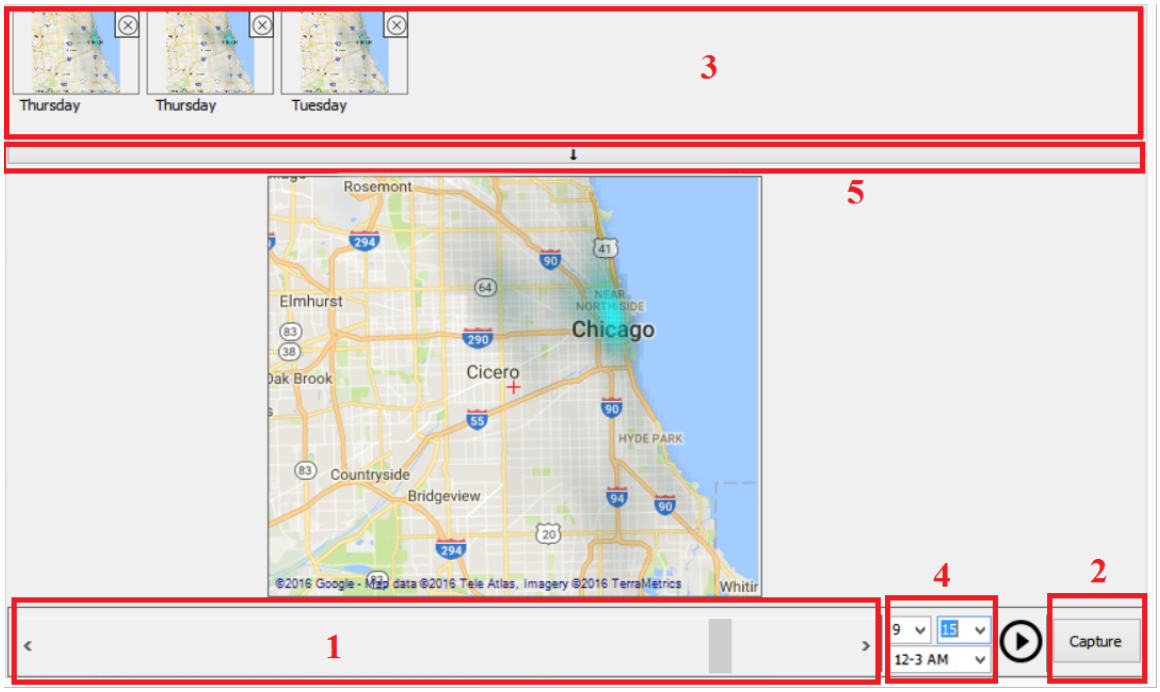


Figure 5.1: A dynamic hotpots visualization of Chicago, Illinois, United States. Users can explore the change in hotspot patterns by using the slide (1) or by going to a specific data and time interval (4). Snapshots can be captured (2) for further side by side comparison (3).

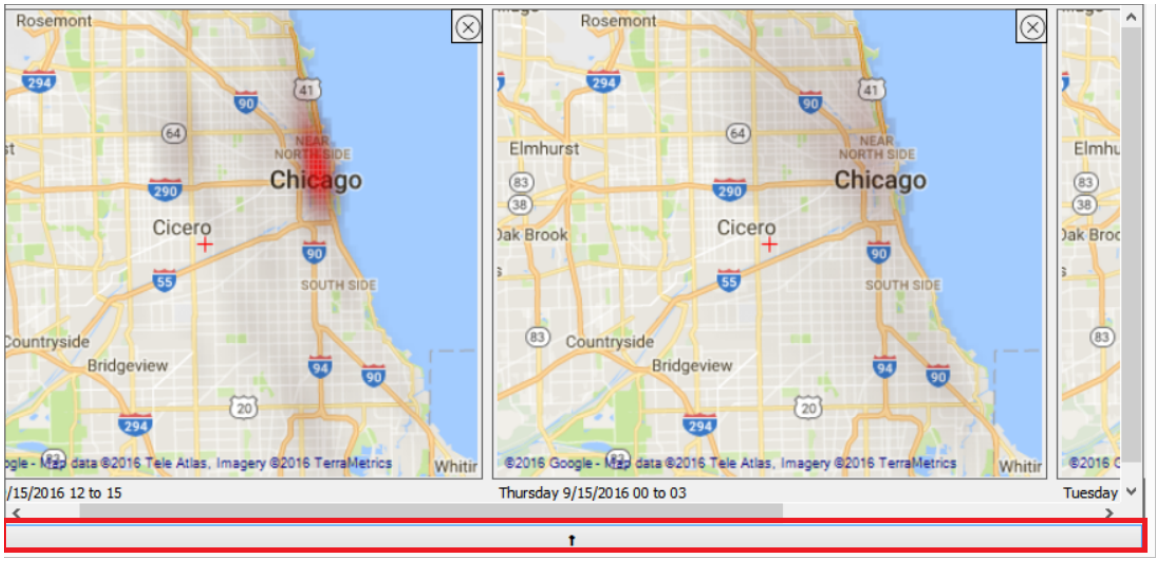


Figure 5.2: Comparing side by side hotspot snapshots of Chicago.

5.3 Experimental design

The human factor experiment consists of the following parts:

- A registration process where the participant is asked to enter their age, gender, occupation (law enforcement, student, or other) whether or not they have any experience in reading spatiotemporal data from a map, the amount of experience (if any), whether or not they have a color vision deficiency, and the type of that deficiency (if any).

- After registration, participants click on the "Start Experiment" button which will assign them randomly to either one of the three designs: 1) Static maps; 2) grid of maps; and 3) dynamic maps.

- First, I provide a description of hotspot maps along with a brief Pre-Test to test the participant understanding of hotspot maps. Also, in the final question of the main part of the experiment, I ask participants to draw a movement route on the screen. In the Pre-Test part of the experiment, I provide instructions on how to draw a route, and I ask participants to redraw a sample route that is being shown on a separate map.

- In case the participant is assigned to Design 2 or 3, I provide extra descriptions explaining the grid of maps or the dynamic visualization. I will further test participants' understanding of dynamic maps' functionalities by asking them to perform three pre-test questions that require them to manipulate the map.

- Since the main experiment requires that participants understand the concept of hotspots and know how to navigate the maps, the experiment's application will not allow the participant to proceed with the experiment without correctly finishing the pre-test questions. The application will provide them with supporting instructions in case they fail to complete the Pre-Test questions.

- The main part of the experiment consists of four questions. In each question, I will provide participants with hotspot maps (static, grid or dynamic) that are generated from actual theft incidents from Chicago, Illinois between 2016-09-17and 2016-09-23.

- In the first question, I will ask the participants to make a spatial decision to choose at most four out of seven given areas with the highest Theft risk. For the second question, I ask the participants to make a temporal decision to choose one out of eight given time intervals with the highest theft risk. Likewise, the decision in the third question is both spatial and temporal such that participants need to choose the areas as well as the time intervals with the highest theft risk. Finally, in the fourth question, I will ask the participants to draw the route between two given police stations such that the route should go through the highest risk areas while not exceeding 18 miles. The length of the shortest route between the two stations is 13.8 miles.

- Post-test questionnaire regarding the participants' experience with the visualization method in terms of ease-of-use and functionalities that the participants would like to see in the visualizations.

Three type of participants were recruited for the experiment: Students, Police officers, and others. Police officers were recruited with the help of crime analysts from the University

of Virginia Police Department,[1] and Albemarle County Police Department.[2] Participants had the ability to run the experiment online since it was provided through an Amazon EC2 Remote Machine.[3]

5.4 Evaluation

I used two evaluation metrics to measure the performance of the three designs: 1) response time which is computed (in seconds) for each question, and 2) number of captured incidents within the chosen areas (questions 1 and 3), the chosen time intervals (questions 2 and 3), and within 100 meters from the key points of the chosen route (question 4). The number of captured incidents serves as an indicator for the quality of decisions. Since response time is considered as an evaluation metric in this experiment, and since participants had the option to run the experiment remotely, I added a "pause" button at each question. Participants can click on the "pause" button to temporarily pause the experiment.

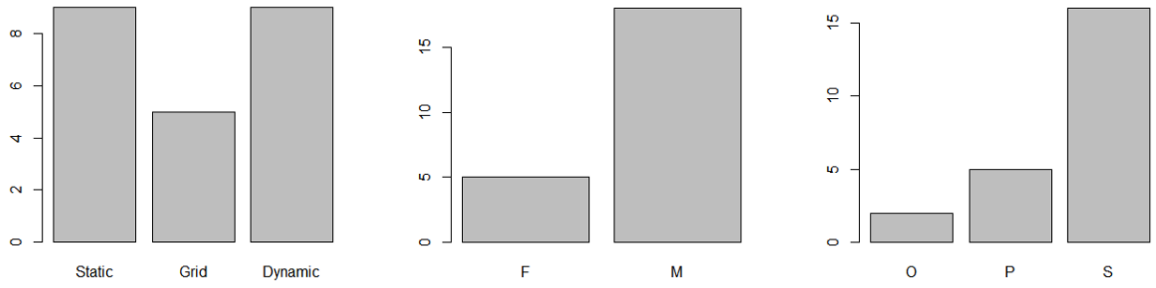


Figure 5.3: Demographics of 23 participants who are mostly males and students.

I recruited 23 participants who were mostly males and students. Figure 5.3 shows the demographics of individuals who ran the experiment. I had equal number of people running the static and the dynamic designs. Fewer people performed the experiment using the grid design. Also, only 5 police officers were recruited to perform the experiment. Next, I analyzed

[1]University of Virginia Police Department, http://www.virginia.edu/uvapolice/.
[2]Albemarle County Police Department, http://www.albemarle.org/department.asp?department=police.
[3]Amazon EC2 Web Services, https://aws.amazon.com/ec2/.

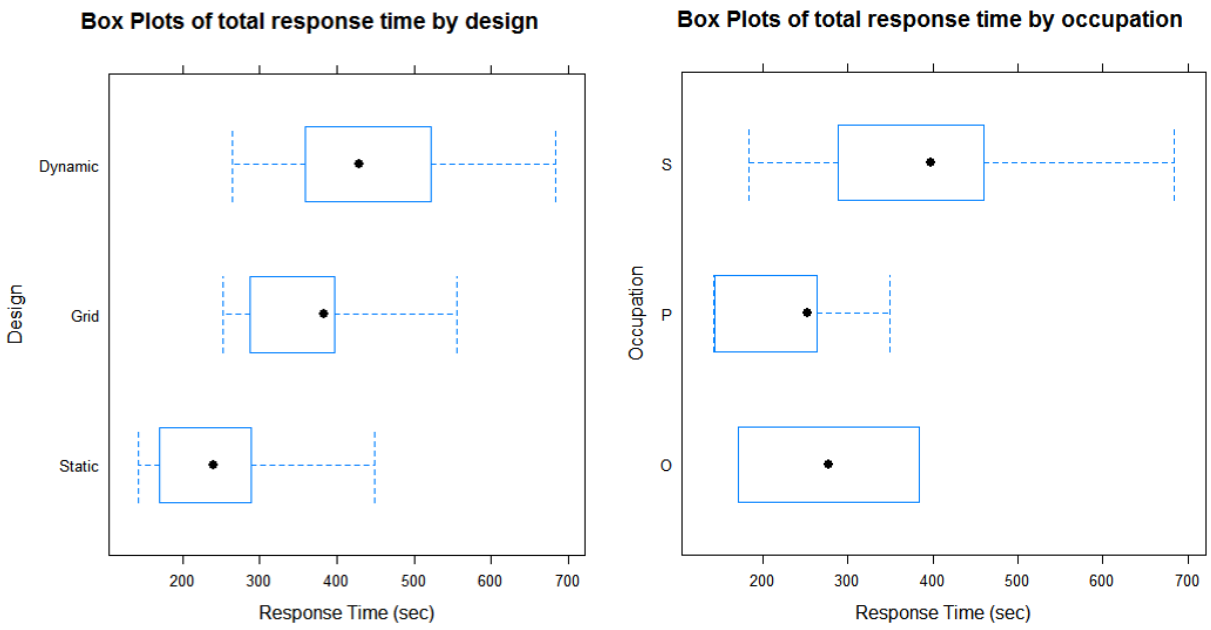


Figure 5.4: Comparing total response time conditioned by design and occupation.

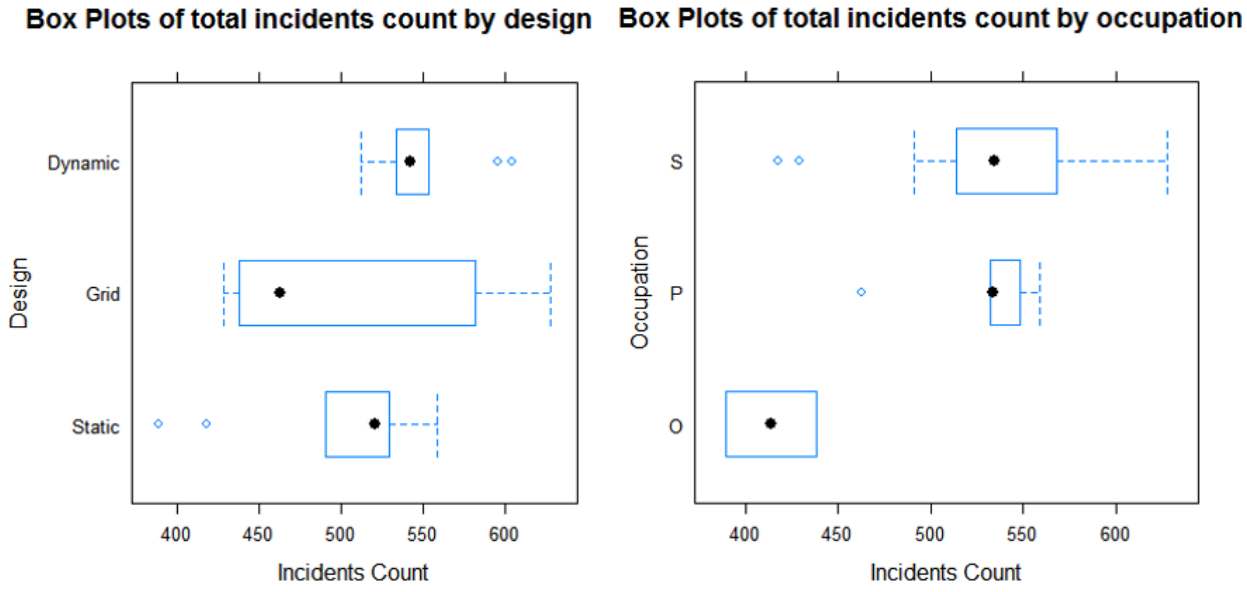


Figure 5.5: Comparing total number of incidents conditioned by design and occupation.

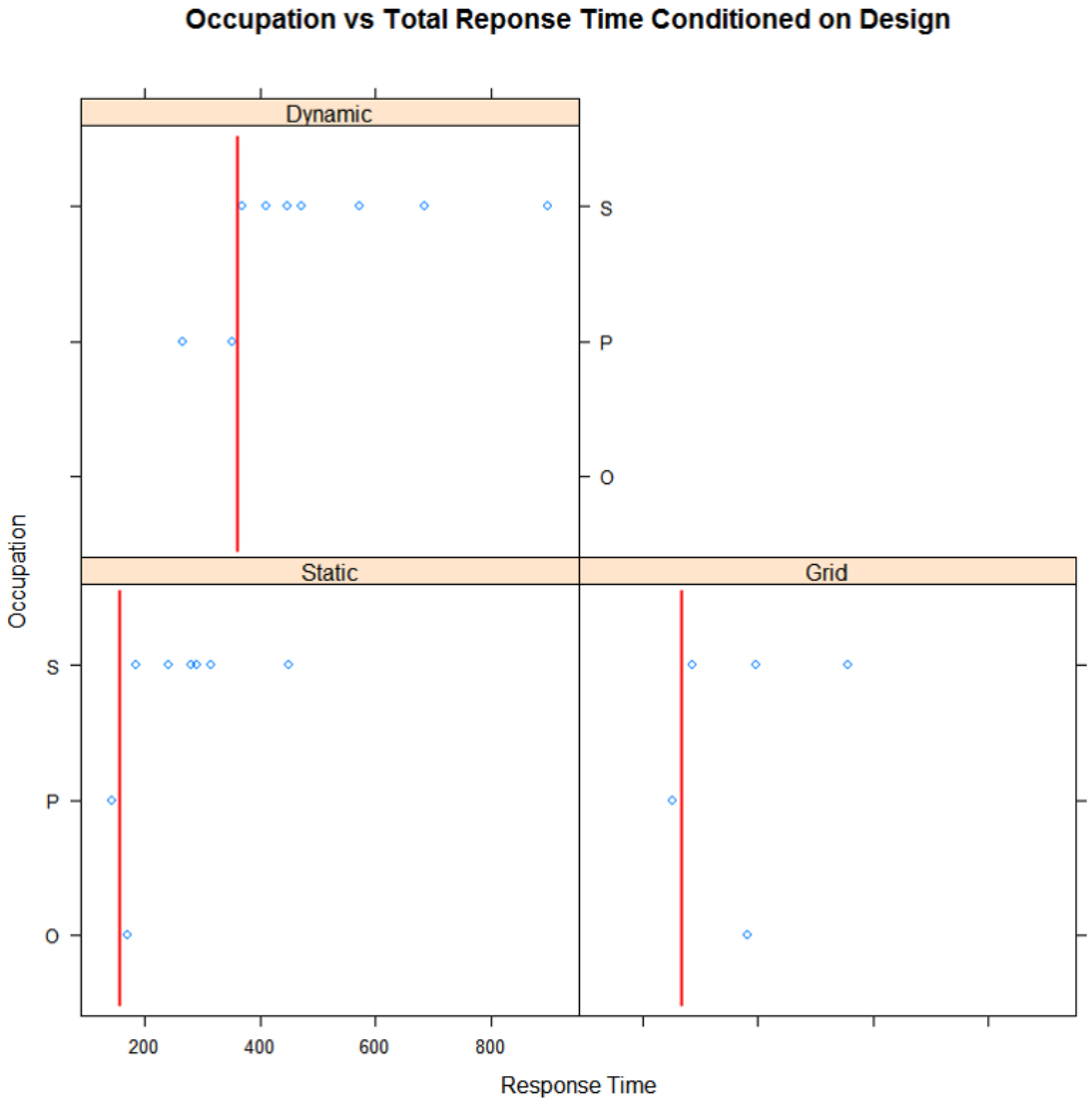


Figure 5.6: Total response time conditioned by both design and occupation.

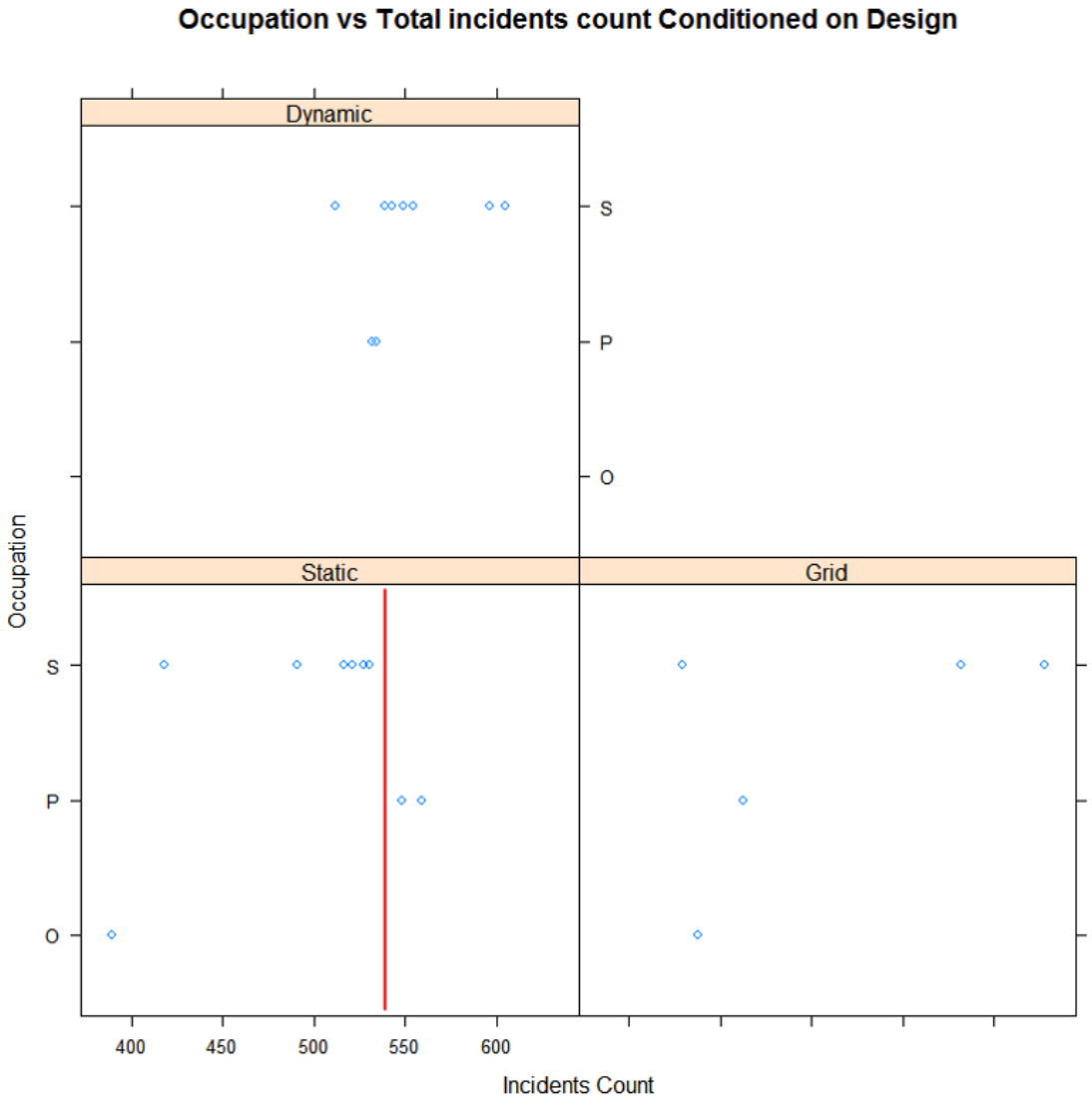


Figure 5.7: Total number of captured incidents conditioned by both design and occupation.

the distribution of participants in terms of their total response time (shown in Figure 5.4) and total number of incidents (shown in Figure 5.5) for the 4 questions using box plots. Static design has the shortest median total response time. Likewise, police participants are the fastest to conclude the experiment. As for the quality of decisions, as measured by the total number of captured incidents, no design has significantly outperformed the other two designs. Furthermore, I used bivariate trellis plots to analyze the performance of participants conditioned on both experiment design and occupation. Figure 5.6 shows the bivariate trellis plot for total response time. Among all participants, police are the fastest regardless of the design. As for the quality of decisions, Figure 5.7 shows that police participants made better decisions than any other participant using the static design.

Table 5.1: Statistical analysis of total response time and number of incidents conditioned on design and occupation.

	Dependent variable:			
	Total Response Time		Total Number of Incidents	
	(1)	(2)	(3)	(4)
Static Design	-250.312^{***}		-51.667^{*}	
	(69.954)		(27.408)	
Grid Design	-120.879		-43.556	
	(82.770)		(32.430)	
Other Occupation		-151.028		-120.250^{***}
		(124.789)		(39.317)
Police Occupation		-197.789^{**}		-6.550
		(85.247)		(26.859)
Observations	23	23	23	23
R^2	0.390	0.234	0.164	0.320
Adjusted R^2	0.329	0.157	0.080	0.252
Residual Std. Error (df = 20)	148.394	166.385	58.142	52.422
F Statistic (df = 2; 20)	6.404^{***}	3.048^{*}	1.959	4.710^{**}
Note:				*p<0.1; **p<0.05; ***p<0.01

Table 5.2: Statistical analysis of number of incidents per question conditioned on design.

	Dependent variable: Number of Incidents Per Question			
	Q1	Q2	Q3	Q4
	(1)	(2)	(3)	(4)
Static Design	4.000	−10.889***	−44.556***	−0.222
	(13.534)	(3.401)	(12.350)	(9.292)
Grid Design	10.844	−10.533**	−14.044	−29.822**
	(16.014)	(4.024)	(14.612)	(10.994)
Observations	23	23	23	23
R^2	0.022	0.379	0.401	0.308
Adjusted R^2	-0.075	0.317	0.341	0.239
Residual Std. Error (df = 20)	28.710	7.215	26.198	19.711
F Statistic (df = 2; 20)	0.229	6.099***	6.701***	4.445**
Note:				*p<0.1; **p<0.05; ***p<0.01

Table 5.3: Statistical analysis of response time per question conditioned on design.

	Dependent variable: Response Time Per Question			
	Q1	Q2	Q3	Q4
	(1)	(2)	(3)	(4)
Static Design	−6.761	−81.909**	−102.899***	−58.743*
	(20.121)	(30.009)	(19.027)	(29.977)
Grid Design	8.581	−52.251	−69.376***	−7.833
	(23.807)	(35.507)	(22.513)	(35.469)
Observations	23	23	23	23
R^2	0.021	0.275	0.600	0.177
Adjusted R^2	-0.077	0.202	0.560	0.094
Residual Std. Error (df = 20)	42.683	63.658	40.363	63.591
F Statistic (df = 2; 20)	0.210	3.787**	15.009***	2.144
Note:				*p<0.1; **p<0.05; ***p<0.01

Box and bivariate trellis plots showed the variations in performance between the three experiment designs. I used linear regression to further test the statistical significance of these variations. Note, since both the designs and the occupations were treated as categorical predictor variables, I choose to use the dynamic design the student occupation as default cases in the encoding of the variables. Table 5.1 shows the linear statistical analysis of total response time and total number of incidents conditioned on design and occupation. The results indicate that: 1) static design is significantly faster than dynamic design, 2) the total number of incidents captured by dynamic design is significantly larger than the one captured by the static design, 3) police participants are significantly faster than students, and 4) participants of the "other" occupation performed significantly worse than students.

The experiment questions were designed to require different types of information to make the decisions. The first question requires only spatial information, the second one requires only temporal information, and both the third and the fourth would require spatio-temporal information. Tables 5.2 and 5.3 show the statistical analysis of response time and number of captured incidents per question conditioned on the design. The results indicate that the response time in making temporal decisions (i.e., question 2) is significantly lower in static and grid designs than dynamic design. Likewise, participants running static and grid designs are significantly faster than the ones running the dynamic design with respect to spatiotemporal decisions (i.e., question 3 for static design, and question 4 for grid design). As for the decision quality, the results indicate that the dynamic design helps in making better temporal and spatiotemporal decisions than the static design because the number of incidents captured by the latter in questions 2, 3, and 4 are significantly lower in static design than dynamic design. Finally, spatiotemporal decisions in question 3 are significantly better in the dynamic design than the grid design.

Finally, the last part of the experiment included a short survey of five questions about the overall user experience. Here are two of the questions and some responses per design and occupation:

- What did you like most about the hotspot visualization methods?
  - **Static - Students:** Simplicity to use.
  - **Static - Police:** location assistance.
  - **Grid - Students:** The blue color was easier to see than other colors I have used in the past.
  - **Grid - Police:** Date and Time were very useful.
  - **Dynamic - Students:** It is easier to understand and visualize as compared to reading numbers. The function to show hot level at different time interval. The dynamic nature of the visual effects make it easy to determine most likely area of interest.
  - **Dynamic - Police:** only helpful if can be Incorporated into current mobile systems.
- What would you suggest to improve the hotspot visualization methods (including additional information that you would like to see)?
  - **Static - Students:** I feel like the software was missing some functionality. For example, I couldn't see any change in the hotspot area over the time intervals, which would have been useful for some questions. Change in the maps by time would be good.
  - **Static - Police:** Time breakout. Good for location but not time.
  - **Grid - Students:** Maps were small and there were too many of them on the screen. I was confused about the many maps that I am looking at. I think this may be due to the connection between the maps is unclear, for example, will it be better if it's not a grid, but just horizontally scrollable with different times of a day and different days all on one dimension.

- **Grid - Police:** Possibly a moving hotspot map so you can scroll through time and watch the crime change visually.
- **Dynamic - Students:** Slow down play option, show how hot is the area by clicking certain point (maybe show a number of some scale). A bigger window to visualize the map. And a more interactive environment with audio help would be extremely helpful. Would like to be able to select something like a summary heatmap of "3-6PM on all days" or "6-9am on weekdays".
- **Dynamic - Police:** Live data. Potential paths to take when routing police. Flagging of days of week where areas have unusual risks.

5.5 Conclusions

The purpose of this study is to compare several methods of visualizing hotspot maps and predicting crime for the purposes of allocating police forces optimally. Practitioners such as police crime analysts, urban city planners, evacuation and rescue squads, etc. have used hotspot maps to make decisions in space and time. However, in most cases, it is hard to perceive the temporal element from hotspots since they convey the information at a fixed point of time. The new methods being tested allow the practitioners to view the map at different times and take snapshots for direct comparisons. I evaluated the performance of participants using response time and number of captured incidents. The statistical analysis shows that the new method helps in making significantly better temporal and spatiotemporal decisions, as measured by the number of captured incidents. However, it requires significantly more time in making such decisions. I anticipated the outcomes of the response time analysis since the new method include more functionalities and practitioners would spend more time navigating and using the tool. Finally, considering that most real case scenarios would involve making spatiotemporal decisions, the new method would be useful for making high quality decisions. Such decisions would improve the resource allocation by the police, and ultimately, help in improving the public safety.

Case study: NIJ Real-Time Crime Forecasting Challenge

The National Institute of Justice (NIJ) have announced a crime prediction competition on September, 1^{st} 2016 titled as: NIJ Real-Time Crime Forecasting Challenge. The competition aimed to compare existing crime prediction systems developed by police agencies, professionals and students. The competition task was to predict different types of incidents and call-for-services (CFS) in the city of Portland, Oregon, United States. Historical crime incidents between 2012-03-01 and 2017-02-28 were provided by the Portland Police Bureau (PPB) over several roll-outs between 2016-09-1 and 2017-02-28. Table 6.1 shows the per type frequency for crime incidents provided by the PPB. Also, the NIJ permitted the use of any external data from either free or fee-based sources such social media, demographics, spatial features, etc. The competition were open for three type of contestants: (1) full-time high school or undergraduate students, (2) small teams or businesses who consist of 20 or less individuals or employees, and (3) large businesses who consist of more than 20 employees. Contestants were asked to submit forecasts for some or all four CFS categories mentioned in Table 6.2.

Crime Type	**Frequency(%)**
Other	783,910 (82.04%)
Street Crime	157,517 (16.48%)
Theft of Auto	9,037 (0.95%)
Burglary	5,096 (0.53%)
Total	955,560

Table 6.1: Frequency of historical crime records in Portland between 2012-03-01 and 2017-02-28.

Beside specifying four CFS categories, the NIJ identified that contestants can submit forecasts to five different prediction windows: (1) One week (2017-03-01 to 2017-03-07),

CFS Category	Code	Translation
Burglary	BURGP	BURGLARY PRIORITY *H
	PROWLP	PROWLER
Street Crime	ASSLTP	ASSAULT PRIORITY
	ASSLTW	ASSAULT WITH WEAPON *H
	DISTP	DISTURBANCE PRIORITY
	DISTW	DISTURBANCE WITH WEAPON *H
	GANG	GANG RELATED
	ROBP	ROBBERY PRIORITY *H
	ROBW	ROBBERY WITH WEAPON *H
	SHOOTW	SHOOTING WITH WEAPON *H
	SHOTS	SHOTS FIRED
	STABW	STABBING WITH WEAPON *H
	THRETP	THREAT - PRIORITY
	THRETW	THREAT - WITH WEAPON *H
	VICE	VICE-DRUGS, LIQUOR, PROSTITUTION, GAMBLING
Theft of Auto	RSTLN	ROLLING STOLEN *H
	VEHREC	VEHICLE RECOVERED
	VEHSTP	VEHICLE STOLEN PRIORITY
All CFS	This category includes all CFS including those in the above categories	

Table 6.2: Four prediction CFS categories which are aggregates of individual sub-categories. Note, *H indicates that the CFS is of high priority which usually involves a weapon.

(2) Two weeks (2017-03-01 to 2017-03-14), (3) One month (2017-03-01 to 2017-03-31), (4) Two months (2017-03-01 to 2017-04-30), and (5) Three months (2017-03-01 to 2017-05-31). Considering the CFS categories and prediction windows, contestants can submit up to 20 predictions.

Submission period was open from 12:00 a.m. (ET) February 22, 2017 through 11:59 p.m. (ET) February 28, 2017, and after crimes have occurred and recorded, submissions will be evaluated using the PAI and PEI* metrics (presented in Chapter 4). The NIJ has announced a total challenge prize of up to $1,200,000 (subject to funds availability) which distributed on winners from the three categories as follows: (1) Large business contestants: 40 prizes of $15,000 each for a total prize of $600,000, (2) Small team/business contestants: 40 prizes of $10,000 each for a total prize of $400,000, and (3) Student contestants: 40 prizes of $5,000 each for a total prize of $200,000. Half of the 40 prizes are awarded to the 20 submissions with highest PAI, and the other half is awarded to the 20 submissions with highest PEI*. Therefore, a winning contestant may win at least one and up to forty different prizes.

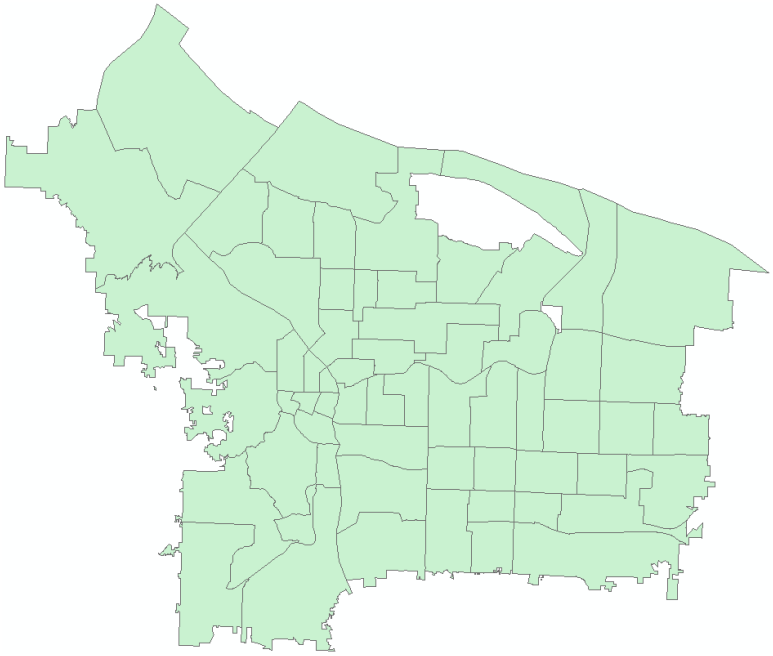


Figure 6.1: Boundary shape-file provided by the NIJ for Portland, Oregon, United States.

6.1 Submission requirements

The NIJ has provided contestants with a boundary shape-file for the prediction area (see Figure 6.1). The NIJ required that each submission adheres to the following requirements: (1) submitted shape-files must contains at least three variables: a unique id for each cell, a binary hotspot variable, and an area variable measured in square feet; (2) cells can be of any shapes, yet individual cell area must be between $62,500 ft^2$ and $360,000 ft^2$, and the minimum cell height is 125 ft; (3) the sum of area of all hotspot cells must be between 0.25 mi^2 and 0.75 mi^2 while the total forecast area need to be equals to 147.71 mi^2 (+/-0.02 mi^2)

6.2 Experimental setup

By the time the submission is open (i.e., 2017-02-22), the NIJ had already released incidents up to 2017-02-14. The following roll-out was on 2017-02-24 which included CFS up to 2017-02-21, and the next one was on 2017-02-27 including CFS up to 2017-02-26. Since I was planning on submitted all 20 forecasts, and considering that only a period of one day between the data roll-out on February 27^{th} and the submission deadline, I decided to train my models on incidents up to 2017-02-21, i.e., I included a lag of 7 days in all crime prediction models.

I experimented with various spatio-temporal models from Chapters 2 and 3. I obtained a number of spatial features such as streets, sidewalks, regional police stations from the Portland open data portal.[1] I also streamed tweets from Portland using Twitter API between November 2016 and February 2017. After exploring various combination of features and models, I found that localized kernel density estimation and multi-kernel density estimation methods performed the best. I built models using one year of historical data starting from 2013-03-01 to 2016-11-30 with three months intervals. For each of the prediction windows, I had 15 different predictions to experiment with. For example, for 1 week prediction window, I trained models to predict CFS for 2013-03-01 $\rightarrow$ 2013-03-07, 2013-06-01 $\rightarrow$ 2013-06-07, ...,

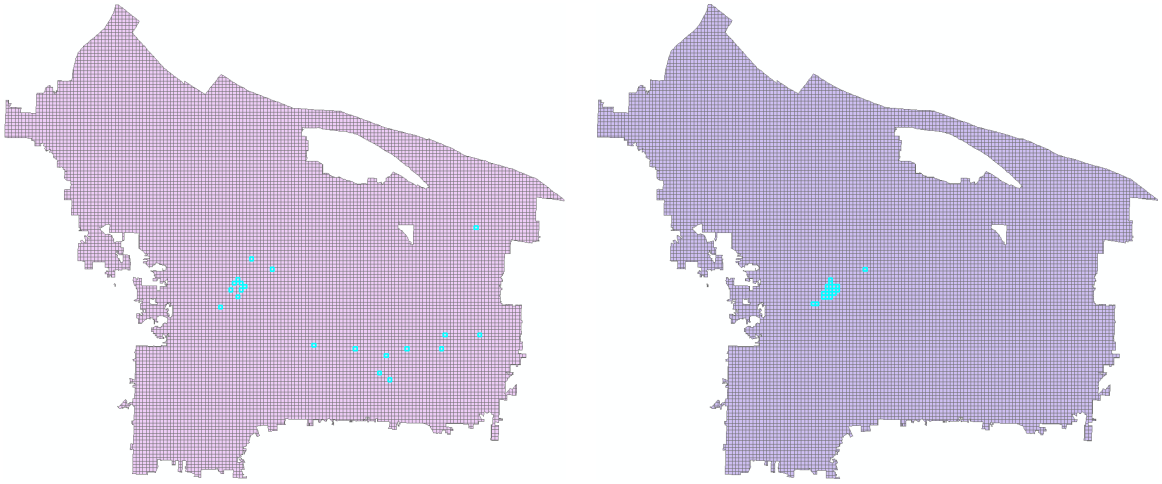
[1]Portland open data portal, http://gis-pdx.opendata.arcgis.com/.

2016-09-01 $\rightarrow$ 2016-09-07. I used a grid of square shaped cells, and I performed a performance analysis (presented in Section 4.3) to decide the best cell size. Considering the submission requirements, mentioned above, I could build grid cells of size starting from 250x250 ft^2 up to 600x600 ft^2. I found that with smaller cells, I would obtain a higher PAI. However, the gap between PAI and PAI* become wider using smaller cells, and therefore, other contestants would have a higher chance in outperforming my models. As for the PEI*, since its formula includes a normalization by the PAI*, my models with larger cells achieved higher PAI PEI*. Therefore, I have decided to use 600x600 ft^2 cells and optimize my models to target higher PEI* values.

Another parameter I had to choose, beside the cell size, is the number of hotspot cells. Since the total area of hotspot cells must be within 0.25 mi^2 and 0.75 mi^2 and using 600x600 ft^2 cells, I could choose at least 20 and up to 58 cells to be hotspots. Considering that CFS categories varies in terms of frequency (i.e., street crimes and all CFS have high frequencies while theft of auto and burglaries have very low frequencies), I decided to focus on two objectives: (1) increasing the average PEI*, and (2) reducing the number of predictions with zero PEI* value. Forecasts on sparse CFS categories (i.e., theft of auto and burglaries) have a significant chance of missing all future incidents (i.e., PAI=PEI*=0) considering that the size of the total area of hotspot cells. For example, a LKDE model forecasting 20 hotspot cells for 1 week prediction window of burglaries (i.e., the sparsest forecast scenario among the 20 different submissions) had zero PEI* value for 8 out of the 15 experimental predictions. I performed an experimental performance analysis and choose the settings, shown in Table 6.3, that would optimize the two objectives.

After building forecasts for the four CFS categories and 5 prediction windows, I evaluated them on the 15 experimental windows mentioned earlier. Table 6.4 shows the average performance metrics including: (1) the total number of incidents (N), (2) the number of incidents occurred within forecasted hotspots (n), (3) the maximum obtainable number of incidents within an area equal to the same area as the hotspots ($n*$), (4) the predictive

accuracy index (PAI), (5) the optimal predictive accuracy index (PAI*), and (6) the predictive efficiency index (PEI*). Figure 6.2 shows two examples of the submitted hotspots for theft of auto (2017-03-01 to 2017-03-31) and all CFS (2017-03-01 to 2017-05-31). These examples illustrates the advantages of localized kernel and multiple localized kernel density estimation methods in dealing with dense and sparse crime types. In sparse crime types, future incidents will less likely to occur in one area, and therefore, methods such as the traditional KDE, which predicts a smooth density surface, will produce a contiguous set of hotspot cells. On the other hand, my approach will produce scattered and localized hotspots cells.



(a) Theft of auto - 1 Month (b) All CFS - 3 Months

Figure 6.2: Forecasts submitted to the NIJ challenge for theft of auto (2017-03-01 to 2017-03-31) and all CFS (2017-03-01 to 2017-05-31). Predicted hotspot cells are highlighted in light blue.

CFS Category		**Prediction Window**				
		1 Week	**2 Weeks**	**1 Month**	**2 Months**	**3 Months**
Burglary	Model	MLKDE	MLKDE	MLKDE	MLKDE	MLKDE
	Cell Size	58	58	58	40	29
Theft of auto	Model	LKDE	MLGKDE	MLGKDE	MLKDE	MLKDE
	Cell Size	58	20	20	20	20
Street crimes	Model	MLKDE	LKDE	LKDE	LKDE	LKDE
	Cell Size	20	20	20	20	20
All CFS	Model	LKDE	LKDE	LKDE	LKDE	LKDE
	Cell Size	20	20	20	20	20

Table 6.3: Models and cell sizes used to predict CFS in Portland, Oregon between 2017-03-01 and 2017-05-31.

CFS Category		Prediction Window				
		1 Week	**2 Weeks**	**1 Month**	**2 Months**	**3 Months**
Burglary	N	149.6	303.5	632.8	1143.46	1565.1
	n	9.7	10.5	27.3	36.8	29.9
	$n*$	56.3	61.73	69.5	73.5	60.3
	PAI	14.53	8.32	9.04	10.72	9.02
	PAI*	97.34	59.86	33.63	25.03	20.58
	PEI*	0.16	0.16	0.37	0.48	0.48
Theft of auto	N	216	418.2	822.8	1486.0	2012.2
	n	21.1	12.0	20.1	28.6	35.8
	$n*$	58.6	29.1	39.2	48.6	57.2
	PAI	22.95	20.75	18.03	15.05	13.57
	PAI*	85.18	67.37	40.28	27.79	23.20
	PEI*	0.34	0.39	0.50	0.58	0.62
Street crimes	N	2367.8	3637.1	5696.2	8772.1	11563.86
	n	58.1	109.1	238.1	477.2	711.53
	$n*$	85.3	138.3	265.5	510.1	748.13
	PAI	17.08	19.59	25.57	31.86	35.56
	PAI*	25.84	25.05	28.53	34.08	37.41
	PEI*	0.67	0.78	0.89	0.93	0.95
All CFS	N	7487.7	11435.4	19816.2	35754.1	51684.7
	n	222.3	428.7	939.0	1888.5	2835.0
	$n*$	261.2	484.8	1015.8	2017.4	2993.3
	PAI	18.07	21.88	27.23	30.19	31.35
	PAI*	21.29	24.83	29.53	32.32	33.17
	PEI*	0.84	0.88	0.92	0.93	0.94

Table 6.4: Average performance metrics for forecasts in Table 6.3 evaluated on 15 prediction windows between 2013-03-01 and 2016-11-30 where N is the total number of incidents, n is the number of incidents occurred within forecasted hotspots, and $n*$ is the maximum obtainable number of incidents within an area equal to the same area as the hotspots.

On June, 9^{th} 2017, the NIJ has released all call for services for the prediction window between March 1^{st} and May 31^{st} 2017. Table 6.5 shows the performance of my models on incidents from the prediction window. Eight out of twenty models achieved a PEI* value more than 80%; Four out of twenty models achieved a PEI* value more than 90%; Only one prediction (namely, two weeks of burglaries) failed to capture any future incidents. The NIJ is expected to announce the winners on June 30^{th} 2017. However, until the submission date of the this document (July 28^{th} 2017), the announcement has not been made yet.

CFS Category		Prediction Window				
		1 Week	**2 Weeks**	**1 Month**	**2 Months**	**3 Months**
Burglary	N	20	41	93	175	268
	n	1	0	1	2	4
	$n*$	20	41	60	42	42
	PAI	9.86	0	2.14	3.27	5.89
	PAI*	197.21	197.21	128.44	68.64	61.82
	PEI*	0.05	0	0.02	0.05	0.1
Theft of auto	N	71	135	273	543	805
	n	3	3	7	18	28
	$n*$	59	25	29	47	58
	PAI	8.33	12.71	14.66	18.96	19.89
	PAI*	163.88	105.91	60.75	49.5	41.2
	PEI*	0.05	0.12	0.24	0.38	0.48
Street crimes	N	629	1205	2680	5352	8480
	n	64	112	253	510	822
	$n*$	91	154	302	572	900
	PAI	58.19	53.16	53.99	54.5	55.44
	PAI*	82.74	73.09	64.45	61.12	60.7
	PEI*	0.7	0.73	0.84	0.89	0.91
All CFS	N	3876	8021	17873	35770	55744
	n	224	485	1104	2218	3451
	$n*$	264	561	1198	2378	3702
	PAI	33.05	34.58	35.33	35.46	35.4
	PAI*	38.95	40	38.33	38.02	37.98
	PEI*	0.85	0.86	0.92	0.93	0.93

Table 6.5: Performance of forecasts submitted to the NIJ for incidents between 03/01/2017 and 05/31/2017 where N is the total number of incidents, n is the number of incidents occurred within forecasted hotspots, and $n*$ is the maximum obtainable number of incidents within an area equal to the same area as the hotspots.

Summary of Contributions and Future Work

7.1 Summary of contributions

7.1.1 Three approaches for building localized crime prediction models

This dissertation has described three different methods for creating prediction models at the local level which can be suitable for different prediction scenarios: 1) a new set of features to quantify the micro-level routine activities from Twitter posts, 2) two area-specific models that allow for fitting distinct coefficients for different areas; and 3) a computationally efficient and new kernel density estimation method that relies solely on local information for producing density estimates.

Besides achieving performance gains, the main advantages of the three localized prediction model approaches is: 1) correlating people movements with crime likelihood. These correlations would provide insights on the type of activities that would increase the chance of victimization; 2) varying features coefficients across areas reflects the real world scenario where different places have unique crime patterns which would result in unique correlations. For example, a global model correlating thefts with the distance from bars would assign a single correlation coefficient (either positive or negative) to the entire study space. While in reality, the correlation would depend on the characteristics of the physical space. For examples, thefts patterns surrounding bars in metropolitan areas where many people and/or policing agents are present would be different from those in rural areas. Area-specific models allow coefficients to vary across different local areas which accounts for unique spatial characteristics of those areas; and 3) producing non-smooth hotspots which allow local areas to have zero

threat (i.e. density estimate) in case their immediate surroundings have zero historical crime records. In the traditional KDE, each and every historical incident will contribute to the density value of each and every area regardless of the distance from that area. The distance has the solo effect of reducing or increasing the contribution of historical incidents that are far or near from a particular area.

The three proposed approaches for building localized crime prediction models have unique requirements: 1) geo-tagged behavioral data such as tweets. Correlating the micro-level movements with crime likelihood would require data for reconstructing and quantifying those movements; 2) a clear definition of a local area. In order for area-specific methods to work effectively, they require practitioners to partition the study area into local areas that would have unique crime characteristics. In my experimentations, I defined an area by the zip-code boundaries. However, zip-codes are used for mailing systems and has no clear connection to crime patterns; and 3) historical crime records. Experimental results showed that my localized kernel density estimation method improved prediction performance on all crime types except the four of the least frequent crime types. These results would suggest that the LKDE method would require a certain number of historical crime records before it attain gains over the traditional KDE. This requirement was later resolved by the multi-localized kernel approach. However, like all density estimation methods, no density calculations can be made in case practitioners have no access to geo-spatial historical crime incidents. Considering the fact that not all of the previous requirements would be satisfied in real world applications, I choose to analyze the three methods separately and show their usefulness in lieu of the other two methods.

7.1.2 Prediction models of practical use for decision making

Security and law enforcement agencies have limited resources to cover only a small percentage of the most threatened areas. The crime prediction models presented in this dissertation, especially the LKDE and MLKDE models, reached performance gains within small surveillance area percentages. For example, LKDE models reached peak gains up to

25.22% only within the most threatened 10% areas for 6 out of 17 crime types in Chicago. Such gains make my models desirable for practical use to solve real-world crime prediction problems. Furthermore, practical gains motivate the use of our models in randomized controlled field trials such as the ones in [72, 73].

7.1.3 A comprehensive analysis of crime prediction evaluation metrics

This dissertation has discussed two widely used evaluation metrics in the crime forecasting domain, namely predictive efficiency index* (PEI*), predictive accuracy index (PAI). The dissertation further presented a performance study analyzing the behavior of these metrics with respect to different problem settings, particularly the grid cell size. The study showed that using bigger cell sizes will reduce the number of captured incidents per hotspot area, and therefore, reduce the value of PAI. However, the gap between the PAI and optimal PAI* will also be reduced which ultimately cause the PEI* to increase. This comprehensive analysis concluded with the recommendation that with respect to a specific crime prediction problems, models must use the same cell size in order to make a fair comparison in which gains are reached because of model advancements in lieu of problem settings. Furthermore, this dissertation introduced PAI and PEI* plots along with the surveillance plots, and recommend their use over a single PAI and PEI* score since the latter depends on a fixed area percentage. However, generalizing metrics to plots would require that models predict a numeric score (e.g., likelihood of crime occurrence, risk, threat, etc.) instead of a binary hotspot value (e.g., 1 for hotspot; 0 otherwise).

7.1.4 A human factor study to assess visualization interfaces

This dissertation followed a holistic approach such that it is not only important to build advanced prediction models and properly evaluate them, but also analyze the way practitioners perceive their outputs and use them to inform decision making. This dissertation included a human factor study to compare and contrast three visualization methods: static maps, grid maps, and newly proposed dynamic maps. Students and police officers used the different

methods to make three types of decisions: spatial, temporal and spatio-temporal. Statistical analysis of response time and decisions quality (as measured by the number of incidents captured within selected areas and/or time intervals) showed that police are significantly faster than students, and decision made using dynamic maps took significantly more time yet had significantly higher quality than the ones made by static maps.

7.2 Recommendations and implications

Practitioners such as crime analysts can use models presented in this work to predict areas with high change of future crimes and allocate their resources to those areas in order to potentially prevent the future incidents. The police can apply these models using the following steps: 1) choose an area of interest (e.g., neighborhood, police district, city, etc.). The only requirement is to know the coordinates of bounding box of that area in order to build a spatial grid on top of it; 2) choose one or more crime types to predict. The only requirement is to have access to historical spatial incidents in order to build training sets (i.e., positive points); 3) choose feature sets and predictive models. This dissertation included a wide range of predictive models and feature sets. However, their effectiveness can change from one area of interest to another and from one crime type to another. Therefore, it is recommended that practitioners try different models and features, and choose the best combination for their particular prediction problem. Nonetheless, when dealing with dense crime types, I found that LKDE models are very effective for producing high quality predictions; 4) choose the appropriate coverage area to evaluate models. Although evaluation plots presented in this work provide performance at all coverage areas, it is important for practitioners to focus in their evaluation on the area coverage that reflects their existing available resources. Also, I highly recommend that all evaluated models be compared using the same problem settings (e.g., grid cell size).

On the other hand, two practical implications or concerns can be results from this work. First, some criminologists have argued that hot spot models, including the ones proposed in this dissertation, don't result in crime reduction. Instead, crime will be displaced from one

hot spot area (with more surveillance focus) to another area with more opportunities) [74–76]. However, experimental design used in this work allow for running short term predictions (e.g., 6 hours, 12 hours, 1 day, etc.), and therefore, if hot spots are to be displaced, then my models can adapt to the displacement by predicting new hot spots as soon as crime patterns change. Second, considering that all parts of this work are provided for public use, then anyone having access to appropriate data (e.g., historical incidents, spatial features, etc.) can build these models and generate predictions. Therefore, motivated offenders can use such models to predict the areas that the police will focus on and target other areas. This concern can be address in various ways. For example, the police can 1) run a combination of models and features to generate predictions, and then aggregate these predictions using ensemble techniques. Therefore, criminals will not be able to generate the same aggregated predictions although they have the capability to build the individual models; 2) follow an exploitation vs exploration strategies (such as an ϵ-greedy approach [77]), and therefore, even if both police and criminals operate using the same hot spot predictions, there is a chance of patrolling new areas which can not be predicted by criminals.

7.3 Summary of future work

The work presented in this dissertation can be extend in various ways and there are a number of areas left open for future improvement:

First, in the micro-level routine activity models (presented in Section 3.2), I quantified individuals' daily movements using bag-of-venues representation. This approach does not take into account the sequence of venue visits over time. Second, users' activities were mapped to the top-level venue categories, shown in Table 3.1. These categories have a high level of abstraction, which might reduce the discriminative properties of my venue-based features. For example, the category Shop & Service includes both car rental businesses and malls. These places are subject to different crime types, but when both places are mapped to one category, the features become less discriminative. Building a predictive model that takes into account the order of the activities and detailed venue classifications could further

improve prediction performance. However, this change will pose new challenges in dealing with high-dimensional feature spaces, which may require additional work on dimensionality reduction or regularization methods for crime prediction.

Second, I plan to investigate the use of area-grouping approaches (e.g. a clustering grid cells based on their feature values) to build spatial hierarchies. This would help in guiding the sharing of information across areas within the area-specific models presented in Section 3.3. This would also highlight similar areas for uniform intervention, thus reducing resource expenditures. Furthermore, similar to the work in [78], I would like to compare and contrast the use both multi-task and hierarchical models for selecting optimal feature sets.

Third, LKDE and MLKDE models presented in Sections 3.4 and 3.5 can be future extended in two ways: (1) personalization methods such as the ones used to build area-specific models can be used to facilitate the learning of different kernels for various sections of the study region; and (2) deep convolutional neural networks can be used to learn the localized convolutional kernels.

Fourth, in Chapter 4, I performed all analyses using a grid of square shape cells. I would like to explore the effect of using different shapes (e.g., triangles, rectangles, hexagons, etc.) on the prediction performance. Also, the grid used in these analyses is uniform, i.e., all cells are of the same size. I would like extend this work to analyze the performance of prediction models that use a non-uniform grid. Moreover, researchers in [79] found that the temporal component can have a significant impact of three evaluation metrics used for scoring district designs. Motivated by their work, I would like to extend this analysis to study the impact of changing the prediction window (e.g., one day to one week) on evaluation metrics.

Bibliography

[1] Matthew Friedman, Ames Grawert, and James Cullen. Crime trends: 1990-2016. *rennan Center for Justice at New York University School of Law*, 2017.

[2] George L Kelling and William J Bratton. Declining crime rates: Insiders' views of the new york city story. *The Journal of Criminal Law and Criminology (1973-)*, 88(4):1217–1232, 1998.

[3] John E Conklin and J Jacobson. Why crime rates fell. *Crime and Justice International*, 19(72):17–20, 2003.

[4] Steven D Levitt. Understanding why crime fell in the 1990s: Four factors that explain the decline and six that do not. *The Journal of Economic Perspectives*, 18(1):163–190, 2004.

[5] John Eck, Spencer Chainey, James Cameron, and R Wilson. *Mapping crime: Understanding hotspots*. National Institute of Justice, 2005.

[6] Joel M Caplan and Leslie W Kennedy. Risk terrain modeling compendium. *Rutgers Center on Public Security, Newark*, 2011.

[7] Hua Liu and Donald E Brown. A new point process transition density model for space-time event prediction. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 34(3):310–324, 2004.

[8] Michael A Smith and Donald E Brown. Discrete choice analysis of spatial attack sites. *Information Systems and e-Business Management*, 5(3):255–274, 2007.

[9] Xiaofeng Wang, Matthew S Gerber, and Donald E Brown. Automatic crime prediction using events extracted from twitter posts. In *Social Computing, Behavioral-Cultural Modeling and Prediction*, pages 231–238. Springer, 2012.

[10] Matthew S Gerber. Predicting crime using twitter and kernel density estimation. *Decision Support Systems*, 61:115–125, 2014.

[11] Samuel H Huddleston and Donald E Brown. A statistical threat assessment. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 39(6):1307–1315, 2009.

[12] Xiaofeng Wang, Donald Brown, and Matthew Gerber. Spatio-temporal modeling of criminal incidents using geographic, demographic, and twitter-derived information. In *International Conference on Intelligence and Security Informatics*, Lecture Notes in Computer Science. IEEE Press, IEEE Press, 2012.

[13] Brian A. Reaves. Local police departments. *Bureau of Justice Statistics*, 12 2010.

[14] Kate J Bowers, Shane D Johnson, and Ken Pease. Prospective hot-spotting the future of crime mapping? *British Journal of Criminology*, 44(5):641–658, 2004.

[15] Spencer Chainey and Jerry Ratcliffe. *GIS and crime mapping*. John Wiley & Sons, 2013.

[16] Sara McLafferty, Doug Williamson, and Philip G McGuire. Identifying crime hot spots using kernel smoothing. *Analyzing crime patterns: Frontiers of practice*, pages 77–85, 2000.

[17] Anthony A Braga. Hot spots policing and crime prevention: A systematic review of randomized controlled trials. *Journal of experimental criminology*, 1(3):317–342, 2005.

[18] Spencer Chainey, Lisa Tompson, and Sebastian Uhlig. The utility of hotspot mapping for predicting spatial patterns of crime. *Security Journal*, 21(1-2):4–28, 2008.

[19] Murray Rosenblatt et al. Remarks on some nonparametric estimates of a density function. *The Annals of Mathematical Statistics*, 27(3):832–837, 1956.

[20] Emanuel Parzen. On estimation of a probability density function and mode. *The annals of mathematical statistics*, 33(3):1065–1076, 1962.

[21] Trevor C Bailey and Anthony C Gatrell. *Interactive spatial data analysis*, volume 413. Longman Scientific & Technical Essex, 1995.

[22] Marcus Felson. Those who discourage crime. *Crime and place*, 4:53–66, 1995.

[23] MP Wand and MC Jones. Multivariate plug-in bandwidth selection. *Computational Statistics*, 9(2):97–116, 1994.

[24] Joel M Caplan and Leslie W Kennedy. *Risk terrain modeling manual: Theoretical framework and technical steps of spatial risk assessment for crime analysis*. Rutgers Center on Public Security, 2010.

[25] Ian H Witten, Eibe Frank, Mark A Hall, and Christopher J Pal. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.

[26] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. Liblinear: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874, 2008.

[27] Marcus Felson. *Linking Criminal Choices, Routine Activities, Informal Control, and Criminal Outcomes*. Citeseer, 1986.

[28] Bamshad Mobasher, Robert Cooley, and Jaideep Srivastava. Automatic personalization based on web usage mining. *Communications of the ACM*, 43(8):142–151, 2000.

[29] Abhinandan S Das, Mayur Datar, Ashutosh Garg, and Shyam Rajaram. Google news personalization: scalable online collaborative filtering. In *Proceedings of the 16th international conference on World Wide Web*, pages 271–280. ACM, 2007.

[30] Hongning Wang, Xiaodong He, Ming-Wei Chang, Yang Song, Ryen W White, and Wei Chu. Personalized ranking model adaptation for web search. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 323–332. ACM, 2013.

[31] Mohammad Al Boni, Keira Qi Zhou, Hongning Wang, and Matthew Gerber. Model adaptation for personalized opinion analysis. In *The 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, Beijing, China, 07 2015. Association for Computational Linguistics.

[32] Lin Gong, Mohammad Al Boni, and Hongning Wang. Modeling social norms evolution for personalized sentiment classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, volume 1, 2016.

[33] Hsiangchu Lai and Tzyy-Ching Yang. A group-based inference approach to customized marketing on the web integrating clustering and association rules techniques. In *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on*, pages 10–pp. IEEE, 2000.

[34] Andriy Shepitsen, Jonathan Gemmell, Bamshad Mobasher, and Robin Burke. Personalized recommendation in social tagging systems using hierarchical clustering. In *Proceedings of the 2008 ACM conference on Recommender systems*, pages 259–266. ACM, 2008.

[35] Scott E Beauchamp and Gus Koumarelas. Customized learning and assessment of student based on psychometric models, January 8 2009. US Patent App. 12/350,958.

[36] Margaret A Hamburg and Francis S Collins. The path to personalized medicine. *New England Journal of Medicine*, 363(4):301–304, 2010.

[37] Jean-Roger Le Gall, Stanley Lemeshow, Ghislaine Leleu, Janelle Klar, Jerome Huillard, Montserrat Rué, Daniel Teres, and Antoni Artigas. Customized probability models for early severe sepsis in adult intensive care patients. *Jama*, 273(8):644–650, 1995.

[38] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2010.

[39] John Blitzer, Ryan McDonald, and Fernando Pereira. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 conference on empirical*

methods in natural language processing, pages 120–128. Association for Computational Linguistics, 2006.

[40] Mohammad Al Boni and Matthew S Gerber. Area-specific crime prediction models. In *Machine Learning and Applications (ICMLA), 2016 15th IEEE International Conference on*, pages 671–676. IEEE, 2016.

[41] Theodoros Evgeniou and Massimiliano Pontil. Regularized multi–task learning. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 109–117. ACM, 2004.

[42] Shih-Ya Kuo, Steven J Cuvelier, Chuen-Jim Sheu, and Jihong Solomon Zhao. The concentration of criminal victimization and patterns of routine activities. *International journal of offender therapy and comparative criminology*, 56(4):573–598, 2012.

[43] Xiaofeng Wang and Donald Brown. The spatio-temporal modeling for criminal incidents. *Security Informatics*, 1, 02/2012 2012.

[44] Mohammad Al Boni and Matthew S Gerber. Predicting crime with routine activity patterns inferred from social media. *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, 2016.

[45] Dashun Wang, Dino Pedreschi, Chaoming Song, Fosca Giannotti, and Albert-Laszlo Barabasi. Human mobility, social ties, and link prediction. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 1100–1108, New York, NY, USA, 2011. ACM.

[46] Pan Hui, Augustin Chaintreau, James Scott, Richard Gass, Jon Crowcroft, and Christophe Diot. Pocket switched networks and human mobility in conference environments. In *Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-tolerant Networking*, WDTN '05, pages 244–251, New York, NY, USA, 2005. ACM.

[47] Injong Rhee, Minsu Shin, Seongik Hong, Kyunghan Lee, Seong Joon Kim, and Song Chong. On the levy-walk nature of human mobility. *IEEE/ACM Trans. Netw.*, 19(3):630–643, June 2011.

[48] A Chaintreau, Pan Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott. Impact of human mobility on opportunistic forwarding algorithms. *Mobile Computing, IEEE Transactions on*, 6(6):606–620, June 2007.

[49] Kyunghan Lee, Seongik Hong, Seong Joon Kim, Injong Rhee, and Song Chong. Slaw: A new mobility model for human walks. In *INFOCOM 2009, IEEE*, pages 855–863, April 2009.

[50] Vittoria Colizza, Alain Barrat, Marc Barthelemy, Alain-Jacques Valleron, and Alessandro Vespignani. Modeling the worldwide spread of pandemic influenza: Baseline case and containment interventions. *PLoS Med*, 4(1):e13, 01 2007.

[51] Duygu Balcan, Vittoria Colizza, Bruno Gonalves, Hao Hu, Jos J Ramasco, and Alessandro Vespignani. Multiscale mobility networks and the spatial spreading of infectious diseases. *Proceedings of the National Academy of Sciences*, 106(51):21484–21489, 2009.

[52] Vitaly Belik, Theo Geisel, and Dirk Brockmann. Natural human mobility patterns and spatial spread of infectious diseases. *Physical Review X*, 1(1):011001, 2011.

[53] Marta C Gonzalez, Cesar A Hidalgo, and Albert-Laszlo Barabasi. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, 2008.

[54] Claudio Castellano, Santo Fortunato, and Vittorio Loreto. Statistical physics of social dynamics. *Reviews of modern physics*, 81(2):591, 2009.

[55] Dirk Brockmann, Lars Hufnagel, and Theo Geisel. The scaling laws of human travel. *Nature*, 439(7075):462–465, 2006.

[56] Chaoming Song, Tal Koren, Pu Wang, and Albert-László Barabási. Modelling the scaling properties of human mobility. *Nature Physics*, 6(10):818–823, 2010.

[57] Bureau of Labor Statistics United States Department of Labor. American time use survey. http://www.bls.gov/tus/charts/#sleep, 2014-09-30. Accessed: 2015-02-25.

[58] Lawrence E. Cohen and Marcus Felson. Social change and crime rate trends: A routine activity approach. *American Sociological Review*, 44(4):pp. 588–608, 1979.

[59] Marcus Felson and Lawrence E. Cohen. Human ecology and crime: A routine activity approach. *Human Ecology*, 8(4):pp. 389–406, 1980.

[60] Gerard Salton, Anita Wong, and Chung-Shu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[61] Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge, 2008.

[62] Andrew Gelman and Jennifer Hill. *Data analysis using regression and multilevel/hierarchical models*. Cambridge University Press, 2006.

[63] Ned Levine. The "hottest" part of a hotspot: comments on" the utility of hotspot mapping for predicting spatial patterns of crime". *Security journal*, 21(4):295–302, 2008.

[64] Gaston Pezzuchi. A brief commentary on "the utility of hotspot mapping for predicting spatial patterns of crime". *Security journal*, 21(4):291–292, 2008.

[65] SP Chainey. Examining the influence of cell size and bandwidth size on kernel density estimation crime hotspot maps for predicting spatial patterns of crime. *Bulletin of the Geographical Society of Liege*, 60:7–19, 2013.

[66] Timothy Hart and Paul Zandbergen. Kernel density estimation and hotspot mapping: examining the influence of interpolation method, grid cell size, and bandwidth on crime forecasting. *Policing: An International Journal of Police Strategies & Management*, 37(2):305–323, 2014.

[67] Linda Shapiro and George C Stockman. Computer vision. 2001. *ed: Prentice Hall*, 2001.

[68] Steve Lawrence, C Lee Giles, Ah Chung Tsoi, and Andrew D Back. Face recognition: A convolutional neural-network approach. *IEEE transactions on neural networks*, 8(1):98–113, 1997.

[69] Mohammad Al Boni and Matthew S Gerber. Automatic optimization of localized kernel density estimation for hotspot policing. In *Machine Learning and Applications (ICMLA), 2016 15th IEEE International Conference on*, pages 32–38. IEEE, 2016.

[70] Mohammad Al Boni and Derek T Anderson. Aggregation of ontology matchers in lieu of a reference ontology. In *Computational Science and Computational Intelligence (CSCI), 2014 International Conference on*, volume 1, pages 353–359. IEEE, 2014.

[71] Joel M Hunt. *Do crime hot spots move? Exploring the effects of the modifiable areal unit problem and modifiable temporal unit problem on crime hot spot stability.* PhD thesis, American University, 2016.

[72] LW Kennedy, JM Caplan, and EL Piza. A multi-jurisdictional test of risk terrain modeling and place-based evaluation of environmental risk-based patrol deployment strategies. *Rutgers Center on Public Security, Newark, NJ*, 2015.

[73] George O Mohler, Martin B Short, Sean Malinowski, Mark Johnson, George E Tita, Andrea L Bertozzi, and P Jeffrey Brantingham. Randomized controlled field trials of predictive policing. *Journal of the American statistical association*, 110(512):1399–1411, 2015.

[74] Marcus Felson and Rachel L Boba. *Crime and everyday life.* Sage, 2010.

[75] Derek B Cornish and Ronald V Clarke. Understanding crime displacement: An application of rational choice theory. *Criminology*, 25(4):933–948, 1987.

[76] John E Eck. The threat of crime displacement. In *Criminal Justice Abstracts*, volume 25, pages 527–546, 1993.

[77] Michael Castronovo, Francis Maes, Raphael Fonteneau, and Damien Ernst. Learning exploration/exploitation strategies for single trajectory reinforcement learning. In *European Workshop on Reinforcement Learning*, pages 1–10, 2013.

[78] Jon Fox, Samuel H Huddleston, Matthew S Gerber, and Donald E Brown. Investigating a bayesian hierarchical framework for feature-space modeling of criminal site-selection problems. In *MAICS*, pages 185–192, 2012.

[79] Yue Zhang, Samuel H Huddleston, Donald E Brown, and Gerard P Learmonth. A comparison of evaluation methods for police patrol district designs. In *Proceedings of the 2013 Winter Simulation Conference: Simulation: Making Decisions in a Complex World*, pages 2532–2543. IEEE Press, 2013.

[80] Richard T. Wright and Scott H. Decker. *Burglars on the job: Streetlife and residential break-ins*. Boston: Northeastern University Press, 1994.

[81] Marcus Felson and Rachel L Boba. *Crime and everyday life*. Sage, 2009.

[82] Adam Sadilek and John Krumm. Far out: Predicting long-term human mobility. In *AAAI*, 2012.

[83] Nathan Eagle and Alex Sandy Pentland. Eigenbehaviors: Identifying structure in routine. *Behavioral Ecology and Sociobiology*, 63(7):1057–1066, 2009.

[84] Anastasios Noulas, Salvatore Scellato, Cecilia Mascolo, and Massimiliano Pontil. An empirical study of geographic user activity patterns in foursquare. *ICWSM*, 11:70–573, 2011.

[85] Katayoun Farrahi and Daniel Gatica-Perez. Probabilistic mining of socio-geographic routines from mobile phone data. *Selected Topics in Signal Processing, IEEE Journal of*, 4(4):746–755, 2010.